Ludwig-Maximilians-Universität München

Speech Recognition based Automatic Earthquake Detection and Classification

Dissertation
zur Erlangung des Doktorgrades
der Fakultät für Geowissenschaften der
Ludwig-Maximilians-Universität München

vorgelegt von
Moritz Beyreuther

am
24. 2. 2011

1. Gutachter: Prof. Dr. Heiner Igel
2. Gutachter: Prof. Dr. Matthias Hort

Tag der mündlichen Prüfung: 27. 6. 2011

Zusammenfassung

Die moderne Seismologie zeichnet die Bodenbewegungen mit einem weltweit verteilten Stationsnetz kontinuierlich auf und gilt damit als datenreiche Wissenschaft. Die Extraktion der im Moment interessierenden Daten aus diesen kontinuierlichen Aufzeichnungen, seien es Erdbebensignale oder Nuklearsprengungen oder ä.m., ist eine Herausforderung an die bisher verwendeten Detektions- und Klassifizierungsalgorithmen. Insbesondere bei der Untersuchung von Erdbeben mit niedrigem Signal-zu-Rausch-Verhältnis und nur wenigen verfügbaren Stationen ist die automatische Extraktion der Beben aus den kontinuierlichen Daten bislang nahezu unmöglich. An aktiven Vulkanen interessiert Wissenschaftler u.a. die automatische Klassifizierung der verschiedenen vulkanischen Erdbebentypen, welche sich in der Häufigkeit ihres Auftretens je nach Aktivitätszustand des Vulkans unterscheiden können. Eine solche automatische Klassifizierung bildet die Grundlage und ermöglicht somit überhaupt erst die automatische Aktivitätszustandsbestimmung des Vulkans.
Um die automatische Extraktion, oder Detektion, und Klassifizierung von Erdbeben zu verbessern, werden in der vorliegenden Arbeit Methoden aus der Spracherkennung verwendet. Die Ähnlichkeit zwischen seismologischen Signalen und Sprache untermauert und rechtfertigt die Verwendung dieser Methoden. Aufbauend auf den Ergebnissen der Dissertation von Matthias Ohrnberger wurden zunächst diskrete Hidden Markov Modelle für die automatische Erdbebenklassifizierung der durch Regen induzierten Seismizität am Hochstaufen Massiv (Bayern) verwendet. Hierzu waren nur geringe Änderungen des bereits existierenden Algorithmuses notwendig. Durch den Erfolg dieser ersten Studie bestätigt, wurde der Diskretisierungsschritt durch die Verwendung mehrdimensionaler Wahrscheinlichkeitsverteilungen ersetzt, welche in der allgemeinen Formulierung der Hidden Markov Modelle definiert sind. Die Praktikabilität dieser Modelle wurde anhand einer automatischen Klassifizierung der vulkanischen Beben am Vulkan Teide (Teneriffa) überprüft. Eine analoge Anwendung der Hidden Markov Modelle zur Klassifizierung der Erdbeben am Hochstaufen Massiv war wegen der dort auftretenden stark längenabhängigen Signale nicht erfolgreich. Um diese durch die existierenden Algorithmen bedingte Limitierung zu umgehen, werden erstmals in der Seismologie sogenannte Hidden semi-Markov Modelle in Kombination mit "Weighted Finite-State Transducers" erfolgreich benutzt. Diese Erweiterung der Hidden Markov Modelle ist jedoch äusserst rechenaufwendig und damit aus Gründen der Performance nur bedingt in "Echtzeit" einsetzbar. Ein "post-processing" bereits existierender Daten ist damit nahezu unmöglich. Deshalb wurde die Diskriminierung längenabhängiger seismischer Signale durch mehrere Parameterbindungen und -schätzungen in den Hidden Markov Modellen approximiert. Das daraus abgeleitete Vorgehen wurde an Da-

tensätzen von geothermisch induzierter Seismizität und von vulkan-seismischen Beben am Mt. Merapi (Indonesien) überprüft.
Die oben beschriebenen Vorgehensweisen führen neue Methoden zur Erdbebenklassifizierung in der Seismologie ein. Ein Herausstellungsmerkmal ist die gleichzeitige und vor allem erfolgreiche Anwendung der Methoden auf mehrere echte, d.h. nicht simulierte, kontinuierliche Datensätze. Die Evaluation der hier entwickelten Algorithmen und Theorien wurde anhand von 10.5 Monaten kontinuierlicher seismischer Daten durchgeführt. Dies beinhaltet insgesamt ca. 13 Millionen Klassifizierungen. Die Ergebnisse der Klassifizierung sind entweder durchwegs vergleichbar oder in bestimmten Anwendungsfällen sogar besser als bisher verwendete Methoden, wie z.B. manuell erstellte Erdbebenkataloge oder “short-time-average/long-time-average” Detektionen.

Contents

1 Introduction

The automatic detection and classification of seismic signals is getting more and more significant since data centers have changed their acquisition and archiving system from recording and archiving single data snippets to analyzing and storing continuous seismic waveforms. The resulting large amount of continuous data constitutes the prerequisite for automatic earthquake detection and classification. Based on this data availability the individual motivation for the aspired automatism is manifold:
1.) One of the most promising fields of application is volcano seismology. It reflects the idea that different types of volcano induced seismic signals are produced by different physical processes within the volcano's feeder system and therefore represent different states of volcanic activity. Thus it allows to automatically monitor the volcano activity based on changes in the number of seismic signal types per time interval.
2.) For seismic networks the performance of an automatic analysis system (e.g. Earthworm or SeisComP3, Johnson et al., 1995; Weber et al., 2007) can be improved by automatically separating signals that matter from those signals that a local service is not interested in (e.g. nuclear test ban treaty or excluding quarry blasts from local earthquake catalogues).
3.) The large amount of available continuous data is impossible to classify by hand. Therefore the seismological community has a strong demand for automatic detection-/ classification methodologies that are able to process considerable amounts of data in a reasonable computing time in order to build up consistent earthquake statistics. Earthquake statistics are necessary for earthquake risk mitigation, fault zone mappings or seismic tomography.

Detection, i.e. the separation of a signal from the background noise which we are not interested in, and classification / pattern recognition have a very long tradition in seismology (e.g. Joswig, 1996) with many different approaches. The varying detection and classification methods used in seismology as well as their advantages and disadvantages are extensively introduced in the subsequent section 2.1. The study presented here is a continuation of an approach of Ohrnberger (2001) who used double stochastic models, known as Hidden Markov Models (HMMs), as a new method, which are a well established technology in speech recognition. From this starting point several major extensions are made: Firstly to generalize the HMMs to the multidimensional case, secondly to analyze the impact of their time dependency to earthquakes and consequently, as a third step, to introduce new algorithms to the field of seismology. These steps required the programming of an extensive amount of framework code which, in a last step, allowed to apply and verify all suggested algorithms with real data.

This thesis is structured in five main chapters (including the appendix). Each chapter presents a paper, which has been published or is submitted with myself as first author.

Chapter 2 is a feasibility study for adopting discrete Hidden Markov Models (DHMMs) to the detection and classification of seismic signals at the Bavarian Earthquake Service. It has been published in *Geophysical Journal International*. The theory is broadly introduced and the algorithm is arranged in order with the commonly used detection and classification systems of seismology. The DHMM approach is based on the theory and programs of Ohrnberger (2001) with minor modifications and extensions like normalized envelope and new post-processing. When applied to data from the Bavarian Earthquake Service the performance is promising compared to a standard trigger. This encouraged us to continue the research and development of HMM.

Chapter 3, as a next step, replaces the discretization involved in the DHMM by the use of multidimensional multivariate probability distributions as defined in the general HMM definition itself. This chapter has been published in *Journal of Volcanology and Geothermal Research*. Basically, the algorithms of Ohrnberger (2001) were replaced with the algorithms of the Hidden Markov Model/Speech Recognition Toolkit (HTK) (Young et al., 2002). For this replacement it was necessary to re-program the complete interface code such that it is possible to apply the HTK on seismological data in a general and flexible way. However, the elimination of the discretization step of the DHMM increased the number of free parameters significantly, making the parameters more prune to over-fitting. The parameters have to be chosen with care, and therefore the HMMs are applied to a relatively simple classification task. The objective is the discrimination of three signal classes on Tenerife: noise, earthquake of volcanic origin inside a 25km zone around Teide and close tectonic earthquakes outside this region. Again the performance of the system when applied to a single station is promising. It reaches the accuracy of the local-, "IGN" earthquake catalogue, which is based on a manual classification using multiple stations.

Chapter 4 applies the proposed HMM to earthquakes of the Bavarian Earthquake Service and has been published in *Nonlinear Processes in Geophysics*. The earthquake classes in the Bavarian Network differ by epicentral distance, which allows an easy verification of whether a classification is correct or not. When adapting the algorithms for the Bavarian Network, the resulting discrimination of `near` (epicenter distance 10km - 100km) and `reg` (epicenter distance 100km - 600km) earthquakes is comparatively bad. Both classes have similar characteristics, the major difference is the duration time of the earthquakes. However, the usually assumed time dependence in the HMM is an unrealistic representation of the duration of `near` or `reg` earthquake snippets as well as earthquake snippets in general. In speech synthesis a similar problem arises: The synthesised words sound unnatural if the duration of the speech snippets is unrealistic. The common solution there are Hidden semi-Markov Models (HSMM), which incorporate more general duration probabilities for speech. Thus, in chapter 4, HSMM are applied to the classification of different earthquake classes in the Mt. Hochstaufen

area in the Bavarian Network. The improvements compared to HMM are significant, the only drawback is that the computation time increases at least by a factor of 10.

Chapter 5 describes the efforts to back-propagate the advantages of the HSMM into the HMM scheme and has been submitted to *Geophysical Journal International*. This step is taken to prevent long computation times when using HSMM. Minimum durations and duration probabilties more similar to Gaussians are introduced by coupling multiple parameters through a clustering approach. The method is applied to a single station for the detection of earthquakes that are induced by geothermal power plants. It shows similar results as a manually adapted trigger based on three station coincidences.
For the calculations of this last and all previous chapters well established program libraries were used whenever possible (Mohri et al., 1997; Ohrnberger, 2001; Young et al., 2002; Zen et al., 2009). Nevertheless significant extensions and programming effort was necessary in order to adapt the HMM methodology to seismology. Most of the newly programmed methods and classes are written in Python and their vast area of application resulted in the decision to release them as a software package called ObsPy.

Appendix A introduces ObsPy: A Python toolbox for Seismology and was published in *Seismological Research Letters*. ObsPy was initiated and developed by myself and Robert Barsch at the beginning of this thesis, throughout which commonly usable methods and classes were constantly released. Together with the contributions of Tobias Megies, Lion Krischer, Yannik Behr and others ObsPy developed to the most used seismology package in Python, with contributions from all over the world (see `http://www.obspy.org`). In this chapter/appendix many short code recipes are given in order to emphasize the advantages of using Python with the ObsPy extension. Especially the examples of how to call a shared C or FORTRAN library directly from Python without any wrapper code in C, FORTRAN or any other meta wrapper language might be interesting for seismologists. It actually allows the scientist to easily incorporate custom compiled code into an ObsPy framework and to use the well and continuously tested infrastructure (see `http://tests.obspy.org`) for file import, file export, visualization or signal processing.

2 Continuous Earthquake Detection and Classification using Discrete Hidden Markov Models

by Moritz Beyreuther and Joachim Wassermann

Published in Geophysical Journal International, 2008, 175(3), 1055-1066

Abstract

We present a novel technique to solve the automatic detection and classification problem of earth tremor in a single step by using Hidden Markov Modelling (HMM). While this technique was originally developed in speech recognition, it already showed great promise when applied to volcano induced seismic signals. We apply the HMM classifier to a much simpler problem, i.e. the detection and distance dependent classification of small to medium sized earthquakes. Using the smaller and possibly not perfect data set of earthquakes recorded with three stations of the Bavarian Earthquake Service enables us to better evaluate the advantages and disadvantages of the proposed algorithm and to compare the results with simple and widely used detection techniques (e.g. recursive STA/LTA). Overall the performance of HMM shows good results in the pre-triggered classification tasks and reasonable results in the continuous case. The application of HMMs is illustrated step by step so it can be used as recipe for other applications. Special emphasize is given to the important problem of selecting the features, which best describe the properties of the different signals that are to be classified.

2.1 Introduction

Detection, i.e. the separation of a signal from the background noise which we are not interested in, and pattern recognition have a very long tradition in seismology (e.g. Joswig, 1996). In summary, we can distinguish detection and pattern recognition into three main fields: detecting weak signals in background noise, identifying seismic phases (automatic phase picking) and the discrimination of different earth tremors by

its generating process (e.g. tectonic earthquakes vs. artificial explosions). Early applications of detection and classification, mostly implemented in a cascade like schema, were part of the developments in monitoring nuclear test ban treaties (Tjostheim, 1981; Joswig, 1990; Hoffmann et al., 1999).
The majority of these algorithms are based on the computation of so called characteristic functions as a first step. These functions are then used for detecting the onset of a phase or of a complete event by using the short term vs. long term average (STA/LTA). Following the detection step the different waveforms are then often compared to templates of known signal classes (Tjostheim, 1981; Joswig, 1990). For solving the problem of phase identification numerous algorithms were developed throughout the last decades (Allen, 1982; Baer and Kradolfer, 1987; Sleeman and van Eck, 1999; Leonard, 2000; Gentili and Bragato, 2006). Most of them are based again on the computation of characteristic functions, which often contain the computation of the seismic envelope and / or its behavior in time. When automatization is the key issue, some algorithms correlate the estimated phases directly with trail locations (binder module in EARTHWORM Johnson et al., 1995).
The most promising field for applying pattern recognition techniques is seen in volcano seismology. This originates from the idea that different types of volcano induced seismic signals are produced by different physical processes within the volcanos feeder system and therefore represent different states of volcanic activity. This idea is reflected in the frequency of number of event types per time interval as part of a volcano monitoring systems. It is also highly desirable to automatize this procedure. The majority of already developed algorithms is based on Artificial Neuronal Networks (ANN) which are mainly applied to pre-triggered data (Langer and Falsaperla, 2003; Gentili and Bragato, 2006; Langer et al., 2006). In this case, features (or in other words characteristic functions) are stored in a so called feature vector (e.g. frequency partition, envelope etc.) and then are used in the framework of ANN to determine the different signal classes.

The technique we propose in this study is based on a single step procedure. We combine detection and classification by using double stochastic models, so called Hidden Markov Models (HMMs), which proved to be successful in speech recognition (e.g. Young et al., 2002). HMMs provide a powerful tool to describe highly variable time series based on a data driven stochastic model and therefore allow a more general description of signal classes. This makes it possible to handle large sets of reference patterns (earthquakes or volcanic signals) as opposed to template based pattern matching techniques (Joswig, 1994). In contrast to classical ANNs, HMMs incorporate the time dependence explicitly in the models and thus provide a more adequate representation of the seismic signals. The biggest advantage of HMMs vs. pre-triggered classification is its application directly to the continuous data stream which enables the detection and classification of events with small signal to noise ratio (SNR). This is especially important in the framework of volcano seismology or micro seismicity studies as it is a priori not clear whether the amplitude of an event is more important for the mechanism (e.g. volcanic activity or landslides) than its rate of occurrence.
Consequently HMMs were first introduced in seismology in the field of volcano seismol-

ogy (Ohrnberger, 2001; Alasonati et al., 2006). While the first author already applied HMMs on continuous seismic data, the latter author describes the technique of so called Hidden Markov Trees on pre-detected volcanic signals. Wassermann et al. (2007) applied the technique of HMMs on a volcanic data set, spanning one year. They showed that the automatic recognition of volcano induced events appears to have the same statistics of event rates as the recognition of an experienced observer doing interactive classification on a daily routine analysis (see Fig. 2.1). Even though this result is en-

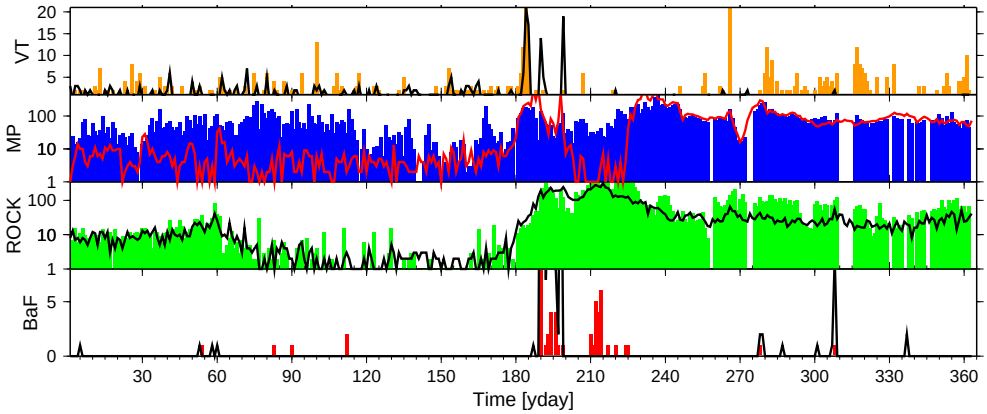

Figure 2.1: Number of visual interactive classified earthquakes per day of the local volcanologists (plotted as bars) versus automatic classification using discrete HMMs (plotted as line graphs). Classification according to the local observatory: VT volcano tectonic, MP multi phase, ROCK rockfall, BaF block and ash flow.

couraging, the evaluation of the classification result is nearly impossible as it includes the interactive reclassification of the whole data set with more than 50 000 detected and classified volcano-seismic events during the time range considered.
Another problem in the evaluation process of volcano induced seismic signals and the assessment of the performance of automatic pattern recognition systems is the assumptions on which the signal separation is based. Only if the separation in e.g. volcanic tremor, hybrid events or volcano tectonic events (McNutt, 2000) is also justified by the true underlying physical process, is it possible to evaluate the overall performance correctly.

In order to circumvent the latter problems, and also because we can increase the performance of the underlying automatic system used within the Bavarian seismic network by separating signals that matters from signals a local service is not interested in the first place, we demonstrate the HMM technique in this paper by applying it to a limited continuous data set of the Bavarian seismic network. In this case the signal or class separation is done by simply using the epicentral distance of earthquakes with respect to the stations used. Therefore it is possible to easily decide whether a classification is correct or wrong.

We next formulate the detection and classification problem in the framework of HMMs. We describe in particular the used features, the properties of the models and explain the solution of the detection and classification problem by adopting HMMs. Finally we evaluate the performance of the one step detection and classification algorithm by comparing its result with the detection rate of a simple recursive STA/LTA trigger.

2.2 Theory

2.2.1 Feature Extraction

As for most classifiers, the time and amplitude dependent seismogram is pre-processed in order to transform it into a domain where the earthquake classes are easier distinguished. This transformation is known as feature generation. Examples of features (or in other words characteristic functions) in seismology are envelope, dominant frequency, power of a frequency band or average absolute amplitude of the seismogram. An experienced seismologist is able to distinguish between earthquake classes "easily", therefore it seems promising to pre-process the seismogram for the classifier by including features which mimic the eye-brain chain: The sharpness in detail can be modelled by detailed features of the time-frequency domain (sonogram); the grosser shape in time, which is recognized by an interaction of eyes and brain, can be mimicked by the seismogram envelope.
The classifier will only perform well if the used features contain different information for the different classes. Therefore the crucial step is to actually find d features in which the classes distinguish themselves most. However, the general formulation of the classifier allows varying configurations of the generated features, and so in the following it is only assumed that some d features have been generated. Detailed explanation of the generation and selection of the features is given in Sec. 2.3.2.

2.2.2 Formulation of the Classification Problem

The classification problem can be expressed as the argument of the maximum probability of the i^{th} class w_i given the observation sequence $\mathbf{O} = \mathbf{o}_1\mathbf{o}_2...\mathbf{o}_T$:

$$\arg\max_i P(w_i|\mathbf{O}). \tag{2.1}$$

With the observation $\mathbf{o}_t$ at time t being either (I) a vector containing the d features in the case of general HMMs, incorporating continuous density models or (II) a symbol $\mathbf{o}_t = o_t$ in case of discrete HMMs (DHMM). In the latter case a standard vector quantifier combined with a pre-whitening transform is used to discretize the feature sequence.
In summary a vector quantifier is crossplotting the training data in a $\mathbb{R}^d$ feature space and segmenting this feature space in such a way that the resulting point clouds are

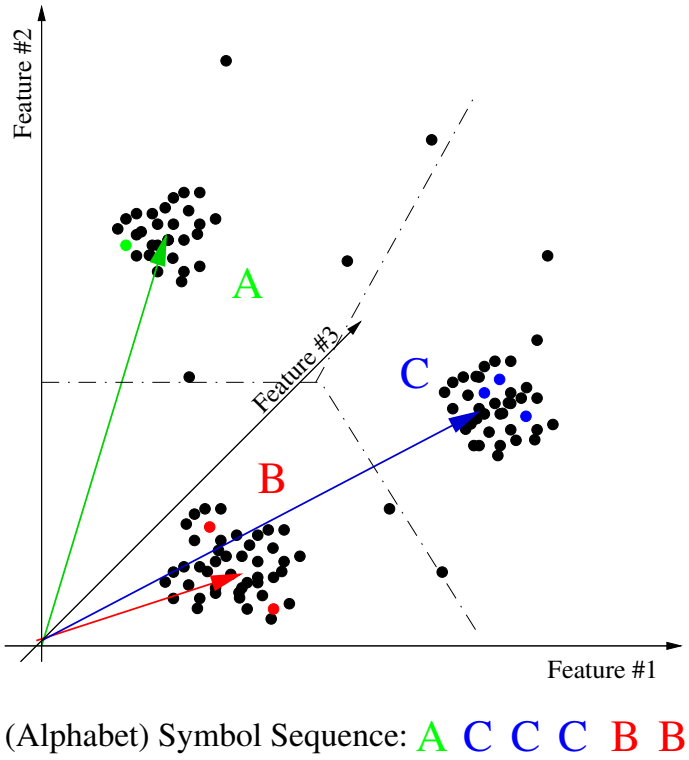

Figure 2.2: Sketch of a vector quantifier: The training data are crossplotted in the feature space resulting in the black points. The feature space is segmented (dashed lines) and labeled by the alphabet symbols A, B, and C such that the point clouds are separated best. A sequence of the green, blue, blue, blue, red and red points is then vectorquantified by assigning the nearest segment, i.e. alphabet symbol, resulting in the (alphabet) symbol sequence A, C, C, C, B and B.

best separated from one another (see Fig. 2.2). Each segment is further labeled with an alphabet symbol (A, B, and C in Fig. 2.2) forming a so called codebook. The vector quantization step is then accomplished by crossplotting a point in the $\mathtt{d}$ dimensional feature space and discretizing the resulting $\mathbb{R}^{\mathtt{d}}$ feature point by assigning the nearest (based on the Euclidean distance) $\mathbb{R}^1$ alphabet symbol. The number of segments, (i.e. number of alphabet symbols, three in case of Fig. 2.2) is further referenced as codebook size. The vector quantifier basically (time independently) pre-classifies the feature vector and therefore enables us to better control the detection and classification system. Despite the fact that information is lost by discretizing the feature vector sequence, we favor the DHMM in order to better control the behavior of the system and used a standard vector quantifier to produce a discrete symbol sequence (for details on VQ see e.g. Ohrnberger, 2001). The remainder of the paper refers to DHMM.
A pre-whitening transform is used before the vector quantification in order to make the feature vectors as independent from each other as possible (in an Euclidean sense).

Therefore the autocovariance matrix of the feature vectors is decomposed by using the singular value decomposition and the resulting eigenvectors are used to transform the feature vectors to the resulting coordinate system (for further details see Kittler and Young, 1973).

Having explained the observation $\mathbf{o}_t$ we return now to Eq. 2.1. $P(w_i|\mathbf{O})$, the probability that the true class is w_i given the observation symbol sequence $\mathbf{O}$, is not directly computed but is calculated from $P(\mathbf{O}|w_i)$, the probability that the observation symbol sequence is $\mathbf{O}$ given the class w_i, by the use of the Bayes's Rule:

$$P(w_i|\mathbf{O}) = \alpha P(\mathbf{O}|w_i)P(w_i); \text{ with } \alpha = \frac{1}{P(\mathbf{O})}. \tag{2.2}$$

$P(w_i)$, the prior probability of class w_i, can either be computed from past events or alternatively be seen as uniform for each class (usual assumption in speech recognition, e.g. Rabiner, 1989). The inverse probability for the observation sequence $\alpha = P(\mathbf{O})^{-1}$ is not depending on the class w_i and therefore is a constant for the maximum calculation of Eq. 2.1.

Given the dimensionality of the problem, the direct estimation of $P(\mathbf{O}|w_i)$ is not feasible. However, if it is assumed that an underlying parametric model such as a DHMM is generating the observation sequence, the problem of estimating $P(\mathbf{O}|w_i)$, the probability that the observation symbol sequence is $\mathbf{O}$ given the class w_i, is replaced by the much simpler problem of estimating $P(\mathbf{O}|\boldsymbol{\lambda}_i)$, the probability that the observation symbol sequence is $\mathbf{O}$ given the parameters of the DHMM $\boldsymbol{\lambda}_i$ (Young et al., 2002).

2.2.3 Parameters of DHMMs

In general, a Hidden Markov Model is a finite state machine, where the states are hidden from direct observation, that changes state once every time unit. Each time t that a state q_t is entered, an observation symbol $\mathbf{o}_t$ is then generated (Young et al., 2002, p. 4).

In detail, a DHMM is described by the parameters:

- $\pi_i = P(q_1 = i)$: the probability that the first state q_1 is i
- $a_{ij} = P(q_{t+1} = j|q_t = i)$: the probability of a state transition from the current state i to the next state j
- $b_j(k) = P(o_t = k|q_t = j)$, the observation symbol probability of the symbol k given the current state is j

Fig. 2.3 shows a sketch of a DHMM, where the hidden states are "unhidden" in order to better illustrate the model parameters. Also for better illustration, we use instead of an arbitrary symbol sequence the number of car accidents per day in Germany 2005 as observation with an average of 6160 accidents per day (Statistisches-Bundesamt, 2006). Consecutive observations during five days $\mathbf{O} = (7219, 7100, 6700, 5980, 6160)$ could be

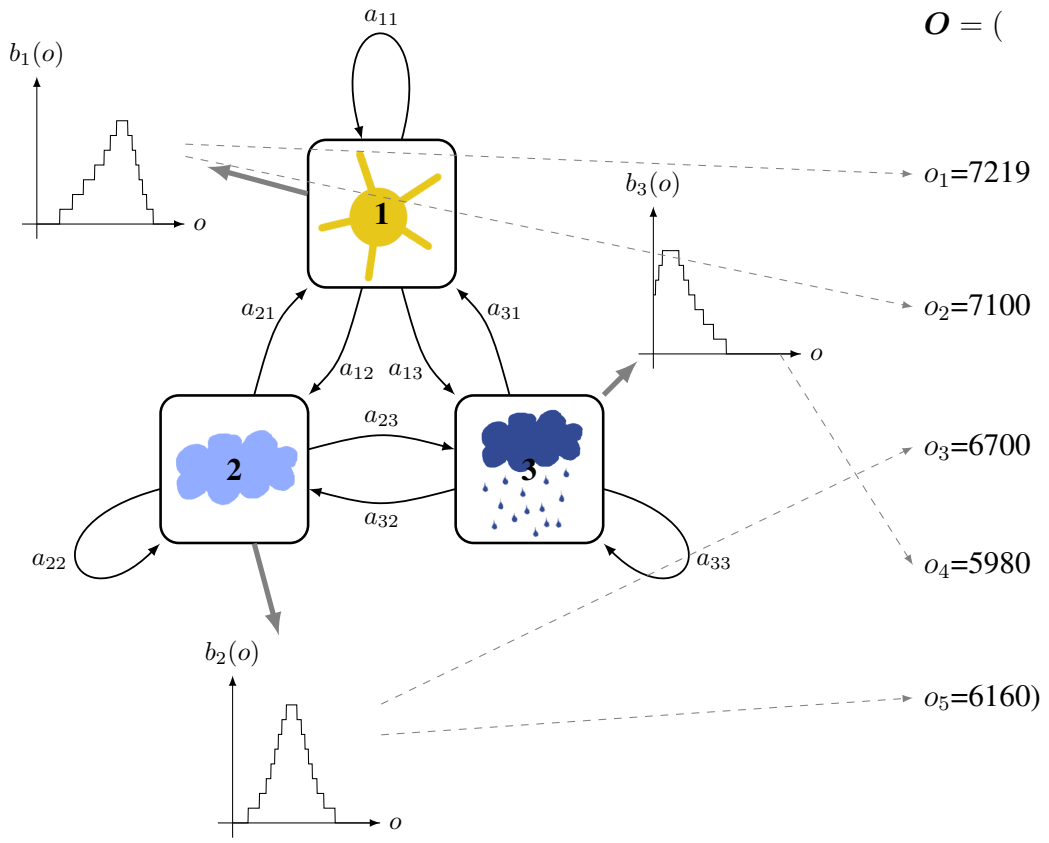

Figure 2.3: DHMM example, which generates a observation sequence $\mathbf{O}$ containing the numbers of car accidents per day in Germany (on average 6160 accidents per day) as realizations of different states of the weather (sun, clouds and rain).

generated by an underlying DHMM, in which the states are different realizations of the weather: sun, clouds and rain. The double stochastic process in this model is composed out of: (I) the transition probabilities of the states (sun, clouds and rain), i.e. the probability of sun tomorrow given that it is cloudy today is $a_{12} = P(q_{t+1} = 1|q_t = 2)$ and the (II) the observation probability (the number of car accidents) which differs in each state, i.e. we assume a high probability of a high number of accidents given sun shine $b_1(\text{"high"}) = P(o_t = \text{"high"}|q_t = 1)$ due to fact that there is denser traffic and therefore it is more probable to have a car accident. The probability of the observation sequence $\mathbf{O}$ and state sequence shown in Fig. 2.3 given model parameters $\boldsymbol{\lambda}$ is calculated by multiplying the temporal evolution of all probabilities which generated the observation sequence:

$$P(\mathbf{O}|\boldsymbol{\lambda}) = \pi_1 b_1(o_1) a_{11} b_1(o_2) a_{12} b_2(o_3) a_{23} b_3(o_4) a_{32} b_2(o_5). \tag{2.3}$$

As mentioned in the example, the probability $P(\mathbf{O}|\boldsymbol{\lambda})$ can be accessed by simply multiplying the relevant DHMM parameters. However, in order to be practicable in real-

word applications there are three basic problems that must be solved for the DHMM (Rabiner, 1989):

Problem 1 Given the observation sequence $\mathbf{O}$ and model $\boldsymbol{\lambda}$, how to efficiently compute $P(\mathbf{O}|\boldsymbol{\lambda})$?

Problem 2 Given the observation sequence $\mathbf{O}$, how to choose the corresponding state sequence $q_1 \dots q_T$ which best explains the observations?

Problem 3 How to adjust the model parameters $\boldsymbol{\lambda}$ in order to maximize $P(\mathbf{O}|\boldsymbol{\lambda})$?

The solution to problem 1 is efficiently given by the so called forward procedure, where dynamic programming is used to avoid redundant calculation. Problem 2 is solved by the Viterbi algorithm which also is based on dynamic programming. An expectation maximization (EM) algorithm called Baum-Welch is used for the solution of problem 3, the training problem. For details see Schukat-Talamazzini (1995); Rabiner (1989); Young et al. (2002).

2.2.4 Continuous Classification and Postprocessing

While the classification of a pre-detected feature symbol sequence $\mathbf{O}$ is given by Eq. 2.1 the real-world detection and classification of a continuous feature symbol sequence is accomplished by cutting feature symbol sequences $\mathbf{O}_{T_i}$ of model dependent length T_i, centered at a midframe, out of the continuous feature symbol stream (see Fig. 2.4). Here frame is used as synonym for window so as to not mix up with the window used for feature generation (Sec. 2.3.2). These resulting partial feature symbol sequences are then evaluated by a normalized log maximum likelihood measure (Juang and Rabiner, 1985; Ohrnberger, 2001):

$$\arg\max_i \frac{1}{T_i} \log P(\mathbf{O}_{T_i}|\boldsymbol{\lambda}_i). \tag{2.4}$$

Moving the midframe along the continuous symbol sequence achieves a classification per frame step.
In order to achieve better performance in this study the following postprocessing steps are applied.
(I) So as to achieve robustness we average the probability of multiple models $\boldsymbol{\lambda}_{ik}$ per class w_i with different number of states $k \in K$:

$$\overline{\log P(\mathbf{O}|\boldsymbol{\lambda}_i)} = \log \sqrt[K]{\prod_{k\in K} P(\mathbf{O}|\boldsymbol{\lambda}_{ik})} = \frac{1}{K}\sum_{k\in K} \log P(\mathbf{O}|\boldsymbol{\lambda}_{ik}). \tag{2.5}$$

The combination of Eq. 2.4 and Eq. 2.5 results in the following "averaged", normalized maximum likelihood measure:

$$\arg\max_i \frac{1}{T_i K}\sum_{k\in K} \log P(\mathbf{O}_{T_i}|\boldsymbol{\lambda}_{ik}). \tag{2.6}$$

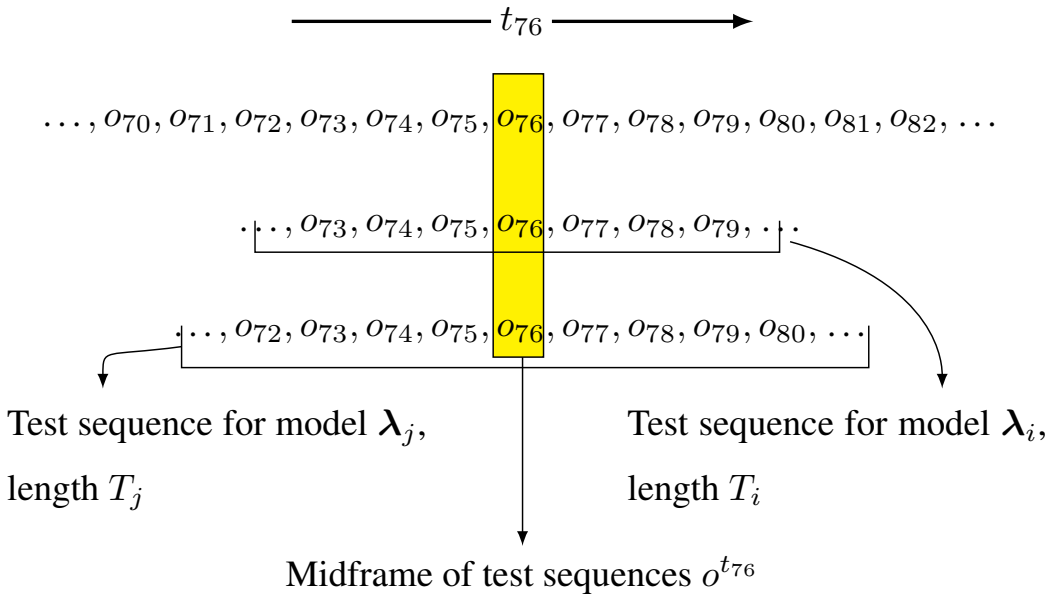

Figure 2.4: Sketch of the continuous classification: Feature symbol test sequences $\mathbf{O}_{\mathsf{T_i}}$ of model dependent length $\mathsf{T_i}$ are selected from the continuous symbol stream shown in the top row around a midframe (colored in yellow) and classified using a normalized likelihood measure (Eq. 2.6) (modified from Ohrnberger 2001).

(II) In the continuous case solely classifications with a three times higher probability than the probability of noise are taken into account. A class dependent minimum event length is used, which is regulating the number of consecutive classifications of one class that must occur such that an event is declared. Furthermore the coincidences of the three stations used in this study are calculated by accepting a classification only if at least two station declare an earthquake classification.

By using the steps explained in this section the earthquake detection and classification algorithm can now be built and is applied in the following.

2.3 Case study

2.3.1 Properties of the Different Classes

In this paper we use real data from three seismological stations from the Bavarian earthquake service to exemplify the steps needed for a DHMM based detection and classification system. The study area is located in a mountainous region where it is of interest to separate local from nearby and from regional earthquakes (Fig. 2.5 shows

example seismograms). The local earthquakes are located within a high mountain and

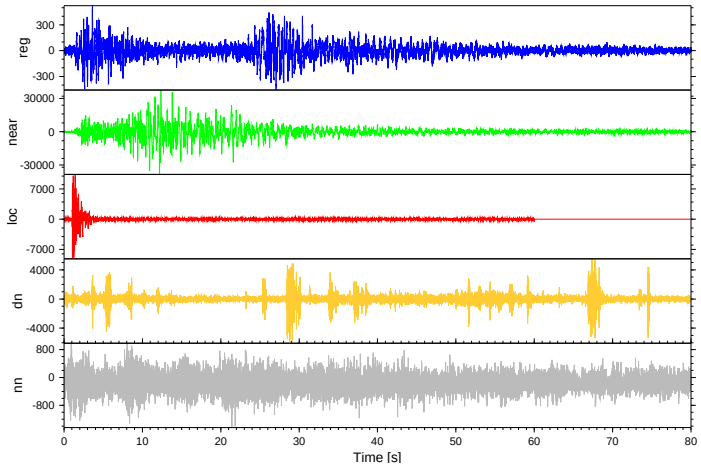

Figure 2.5: Common seismogram for the classes `reg`, `near`, `loc`, `dn` and `nn` (for class definition see Tab. 2.1)

the corresponding wave fields are therefore strongly affected by topography. In contrast, the nearby and regional earthquakes are less affected by topography due to the deeper traveling paths and longer wavelengths involved. This is also expressed by their more similar overall appearance regarding their seismogram.
The classification is crucial for further localizing by using CPU intensive non linear techniques for the local earthquakes (3D-topographic effects) and more classical techniques for the nearby and regional earthquakes. The classes in this case study are identified and defined in Tab. 2.1. Noise in this context is referred to as events of

Type	Epicenter Dist.	Freq. Band	Len.	Charact.	# Events
`loc`	0km - 10km	6Hz - 30Hz	3s	clear P Phase	24
`near`	10km - 100km	2Hz - 15Hz	26s		24
`reg`	100km - 600km	0.5Hz - 8Hz	51s	solid P,S Phase	25
`nn`	night noise	0.3Hz - 30Hz		non anthropogenic	30
`dn`	day noise	0.3Hz - 30Hz		anthropogenic	39

Table 2.1: Characteristics of the different classes / training data. Freq. Band for frequency band, Len. for length, Charact. for characteristics and # Events for number of events in the training data set.

no interest, so that each part of the seismogram can now be associated with one of the above classes. The noise is further divided into two subclasses, day noise `dn` and night noise `nn`. `nn` events are mostly non anthropogenic i.e. metrological influences, vibrating trees, or similar sources known as ambient noise. In contrast `dn` events are mostly anthropogenic, e.g. traffic, logging or saw mills.

Now that the definitions of the classes are given, the next step is to extract features by which the classes can be distinguished from one another so that the classification problem is easier to solve.

2.3.2 Feature Extraction and Selection

The determination of the different classes is not directly based on the seismogram itself but on features extracted from the seismogram (see Sec. 2.2.1). In order to process the features in near real time, a 3s sliding window with step size 0.05s is used for computation. Therefore frequencies are nominally resolved down to $1/3s = 0.33$Hz.
In the following a short background information regarding the calculation of the used features is given. This includes a measure of the classes discriminative power for the different features based on Figs. 2.6, 2.7, and 2.8. The figures show for each feature and for each class respectively the median and histogram over all events in the class dependent training data sets (for number of events see Tab. 2.1).
If the features separate the classes in the median or the histogram distribution, i.e. if the graphs for the different classes do not overlay or in other words are separated, the discriminative power is assumed to be high.

Sonogram Calculation

In order to model the human analyst, the feature of the short term Fourier transform STFT (sonogram) introduced by Joswig (1994) is used. The STFT is smoothed by binning (from k_{low} to k_{high}) the squared complex short time spectrum $Z(\omega_k)$ of the seismogram in n half octave bands (hob_n) (Ohrnberger, 2001, p. 78):

$$hob_n = \ln \frac{\sum_{k=k_{low}(n)}^{k_{high}(n)} |Z(\omega_k)|^2}{\sum_{k=k_{low}(1)}^{k_{high}(N)} |Z(\omega_k)|^2}. \tag{2.7}$$

The following gradation is used in this study; hob_1 0.47-0.78Hz, hob_2 0.70-1.17Hz, hob_3 1.05-1.76Hz, hob_4 1.58-2.63Hz, hob_5 2.37 - 3.95Hz, hob_6 3.56-5.93Hz, hob_7 5.33-8.89Hz, hob_8 8.00-13.33Hz, hob_9 12.00-20.00Hz and hob_{10} 18.00-30.00Hz. In Fig. 2.6, the hob features separate the classes well for both, the median and the histogram distribution.

Polarization Analysis

The polarization properties of the three component seismogram $\mathbf{X} = ((x_{Z1}, .., x_{ZT'}); (x_{N1}, .., x_{NT'}); (x_{E1}, .., x_{ET'}))$ are calculated from the eigenvalue $(\lambda_1 \geqslant \lambda_2 \geqslant \lambda_3)$ decomposition of the covariance matrix $\mathbf{C} = \mathbf{X}^T\mathbf{X}/T'$ of length T': $(\mathbf{C} - \lambda_l^2 \mathbb{1})\mathbf{u}_l = 0$. The features rectilinearity $rect$, the planarity $plan$, the azimuth ϕ_P

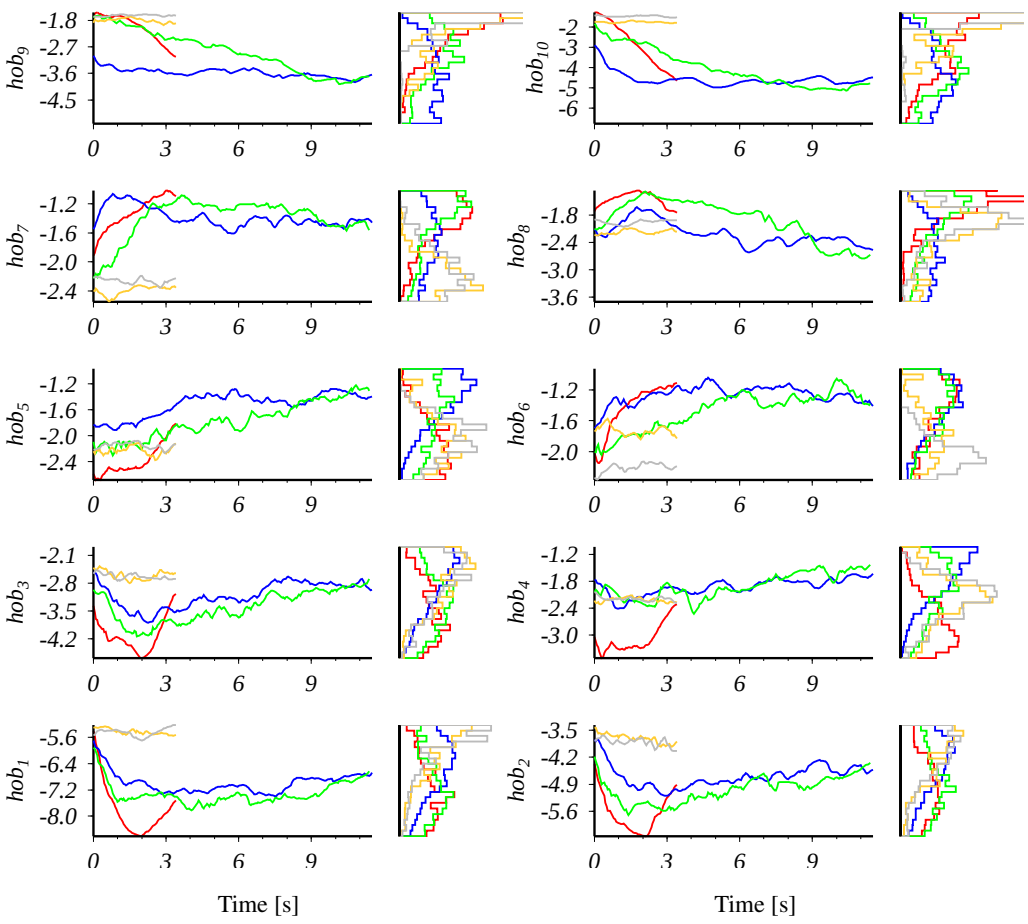

Figure 2.6: The figures show for each feature and for each class respectively the median and histogram over all events in the class dependent training data sets. The features here are $\mathtt{hob}_1 - \mathtt{hob}_{10}$ Color-coding: `reg` in blue, `near` in green, `loc` in red, `dn` in yellow and `nn` in gray.

and the incidence ϑ_P are then computed by using the following equations (Kanasewich, 1981, p. 336f):

$$\mathtt{rect} = 1 - \frac{\lambda_2 + \lambda_3}{2\lambda_1} \tag{2.8}$$

$$\mathtt{plan} = 1 - \frac{2\lambda_3}{\lambda_1 + \lambda_2} \tag{2.9}$$

$$\phi_P = \arctan \frac{u_{1_E}\,\mathrm{sign}(u_{1_Z})}{u_{1_N}\,\mathrm{sign}(u_{1_Z})};\ \mathrm{sign}(x) = \begin{cases} 1, \text{if } x > 0, \\ 0, \text{if } x = 0, \\ -1, \text{if } x < 0, \end{cases} \tag{2.10}$$

$$\vartheta_P = \arccos |u_{1_Z}|. \tag{2.11}$$

While the median of the azimuth in case of `loc` events in Fig. 2.7 separates well

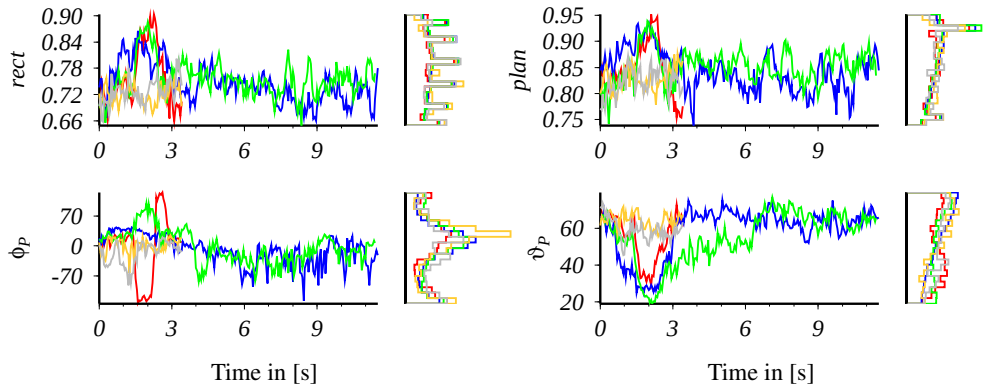

Figure 2.7: The figures show for each feature and for each class respectively the median and histogram over all events in the class dependent training data sets. The features here are rectilinearity $\mathtt{rect}$, planarity $\mathtt{plan}$, azimuth ϕ_P and incidence θ_P. The spikes in the histogram of $\mathtt{rect}$ and $\mathtt{plan}$ are due to the number of bins which is constant for all histograms and not optimal for $\mathtt{rect}$ and $\mathtt{plan}$. Color-coding as in Fig. 2.6.

from all the other classes and while there is a difference visible in the median of the incidence between earthquakes and noise, the features of the planarity and rectilinearity do not show clear separation, neither in the median distribution nor in the histogram distribution.

Complex Trace Analysis

The Hilbert transform $x_{H_i}(\omega) = \mathrm{sign}(\omega)x_i(\omega)$ of the i^{th} component of the Fourier transformed seismogram $x_i(\omega)$ is used for the calculation of the complex trace $x_i^{(a)}(t) = x_i(t) + x_{H_i}(t)$ and its associated features. The envelope $A_i(t)$ and phase $\theta_i(t)$ (for details see Taner et al., 1979; Kanasewich, 1981, p. 367f) are defined as follows:

$$A_i(t) = \sqrt{x_i^2(t) + x_{H_i}^2(t)} \tag{2.12}$$

$$\theta_i(t) = \arctan \frac{x_{H_i}(t)}{x_i(t)} \tag{2.13}$$

$$\approx \int \frac{x_i(t')\frac{d}{dt'}x_{H_i}(t') - x_{H_i}(t')\frac{d}{dt'}x_i(t')}{2\pi x_i x_{H_i}} dt'. \tag{2.14}$$

By using the definition of the envelope and phase, the instantaneous frequency f_i, the instantaneous bandwidth σ_i^2, the normalized envelope $\hat{A}_i$, the centroid time $\mathtt{centr}_i$ and the instantaneous phase difference θ_{ij} can be calculated as follows (for details see

Taner et al., 1979; Ohrnberger, 2001):

$$f_i = \frac{x_i(t)\frac{d}{dt}x_{H_i}(t) - x_{H_i}(t)\frac{d}{dt}x_i(t)}{2\pi x_i x_{H_i}} \tag{2.15}$$

$$\sigma_i^2 = \left(\frac{\frac{d}{dt}A_i(t)}{2\pi A_i(t)}\right)^2 \tag{2.16}$$

$$\hat{A}_i = \exp\left(const \int \frac{\frac{d}{dt}A_i(t)}{f_{Ny}\bar{A}_i}dt\right);\ \bar{A} \text{ smoothed envelope } A \tag{2.17}$$

$$centr_i = \arg\min_t \frac{abs(\int_0^t A_i(t)dt - 0.5\int_0^{T'} A_i(t')dt')}{T'} \tag{2.18}$$

$$\theta_{ij} = \theta_i - \theta_j. \tag{2.19}$$

The definition of the newly introduced normalized envelope (Eq. 2.17) is justified as follows: The smoothed instantaneous bandwidth is normalized by the Nyquist frequency and is integrated in order to receive a normalized measure of the signals envelope. To enlarge the variance of that new measure, a constant is multiplied and the exponent is taken.
The also non standard feature centroid time is the time instance in the 3s sliding window, where 50% of the area below the envelope is reached (Ohrnberger, 2001).
This feature is another type of a normalized envelope measure and performs well for the separation of the earthquake classes in the centroid-time time space (see Fig. 2.8). Good separation in Fig. 2.8 is also obtained by the normalized envelope and the instantaneous bandwidth. Also the instantaneous frequency shows good separation in the histogram distribution while the phase difference shows little separation capabilities.
Having extracted the features, it is now crucial to find and add only those d features to a DHMM detection and classification which separate the different classes best. Adding features which do not separate the different classes will result in a reduced performance of the vector quantifier and consequently affect the following classifier.

In this study we visually select the features with the best discriminative power using Fig. 2.6, 2.7, 2.8. These features are then combined to feature collections (fv) with different total numbers of features, see Table 2.2. Given a feature collection and a

Notation	**Corresponding features**
fv11	$hob_3 - hob_{10}, \hat{A}_z, \sigma_z, centr_z$
fv16	$\phi_P, \vartheta_P, hob_1 - hob_{10}, \hat{A}_Z, \sigma_Z, f_Z, centr_Z$
fv25	$hob_1 - hob_{10}, \hat{A}_{Z,N,E}, \sigma_{Z,N,E}, f_{Z,N,E}, centr_{Z,N,E}, \theta_{NZ,NE,ZE}$
fv29	$\phi_P, \vartheta_P, hob_1 - hob_{10}, \hat{A}_{Z,N,E}, \sigma_{Z,N,E}, f_{Z,N,E}, centr_{Z,N,E}, \theta_{NZ,NE,ZE}$

Table 2.2: Structure of feature collections (for feature definitions see Sec. 2.3.2).

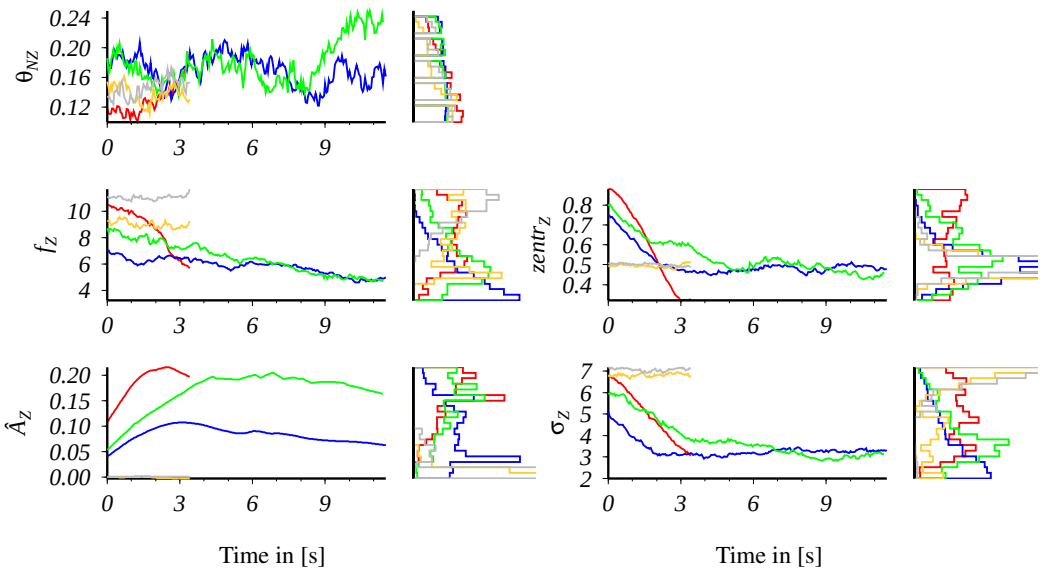

Figure 2.8: The figures show for each feature and for each class respectively the median and histogram over all events in the class dependent training data sets. The features here are phase difference θ_{NZ}, instantaneous frequency f_Z, centroid time $\mathtt{centr}_Z$, normalized envelope $\hat{A}$ and instantaneous phase σ_Z. The zeros in the histogram of θ_{NZ} are due to the number of bins which is constant for all histograms and not optimal for θ_{NZ}. Color-coding as in Fig. 2.6.

training data set (see Tab. 2.1) features are generated from the training data, pre-whitened, vector quantified (64 codebook size, recall Sec. 2.2.2) and used as input for the next step where DHMMs are trained for each class.

2.3.3 Classifier and System Performance

As the number of states is unknown it is beneficial to average over multiple HMMs with different number of states. The number of states in a HMM (e.g. 3 in Fig. 2.3) should be as small as possible in order to estimate robust parameters especially in the presence of a small training data set. On the other hand the model should be complex enough, i.e. have a large enough number of states, to describe the appropriate class characteristics. Being also bound by computational processing time three DHMMs per class are trained with a total number of 3, 4 and 5 states.
After having trained the various DHMMs the detection and classification system is completed. The following approaches are then used to determine the performance of the trained DHMMs, i.e. the classifier:

(I) Usually the performance is tested on single, pre-detected events applying Eq. 2.5.

In speech recognition this is done on an independent pre-labeled (class memberships are known) test data set, which was previously separated from the training data set. In our case there is just a limited amount of training data available. Therefore it is not possible to further reduce the size of the training data set by separating a test data set. Consequently, as a first check, the training data are reevaluated. This test is not impartial as the classifier is evaluated with the same events which are used for training. However, due to the low computational costs this method is interesting for evaluating DHMMs with different feature collections and selecting the one with the best discriminative power.
Fig. 2.9 exemplifies the results of four reevaluations of the training data using different feature collections. The figure shows the true class against the classified class. In order to evaluate the best feature collection, the different types of errors and their corresponding error weight need to be discussed.
A true noise event which is classified as earthquake is called false alarm; a true earthquake which is classified as noise is called missed event and a true earthquake which is classified as a different earthquake type is called confused event. Since reclassifying the whole data set is the only possibility to correct for a missed event, it is common to put the error weight of a missed event higher than of a false alarm. Indeed, a false alarm can be easily corrected by re-verifying only the already classified events. In case of the confused event there is no need for sorting out any false alarms and consequently the lowest weight is put on this kind of error.
Applied to Fig. 2.9 we get the following results: There are no missed events in feature collections fv11, fv16, fv25 and fv29 at all. Fv11 and fv16 are the only ones which have no false alarm and they both have the same classification rates (diagonal elements) for the earthquake classes; only in the classification of the noise classes fv16 performs slightly better (see Fig. 2.9).

(II) A computational more expensive test is the "leave one out" method. Here an independent test data set is simulated by removing one event from the training data set, training the DHMMs from the remaining events and finally classifying the removed event with these DHMMs (Eq. 2.5). This procedure is repeated in a loop over all of the events of the training data set, thus achieving a better measure of the classifiers performance.
Fig. 2.10 shows the results of the "leave one out" method in the form of a confusion matrix. Here, in all feature collections, 6% of the `reg` events are missed. The position of the false alarms are marked in Fig. 2.10 by a red rectangle. Fv11 has together with fv16 the lowest false alarm rate of 12%. Focusing on the correct classifications (sum of diagonal elements) fv11 performs better and is consequently used (in agreement with the previously described evaluation criterions, see Sec. 2.3.3, I) for the following continuous classification.

(III) Since the "leave on out" method is based on single pre-detected events with certain length, continuous data are a more realistic performance test for the detection and classification system. E.g. in only one hour $60 \cdot 60\mathrm{s}/0.25\mathrm{s} = 14400$ (frame rate 0.25s) symbol sequences are classified, thus giving a better view of the DHMM detection

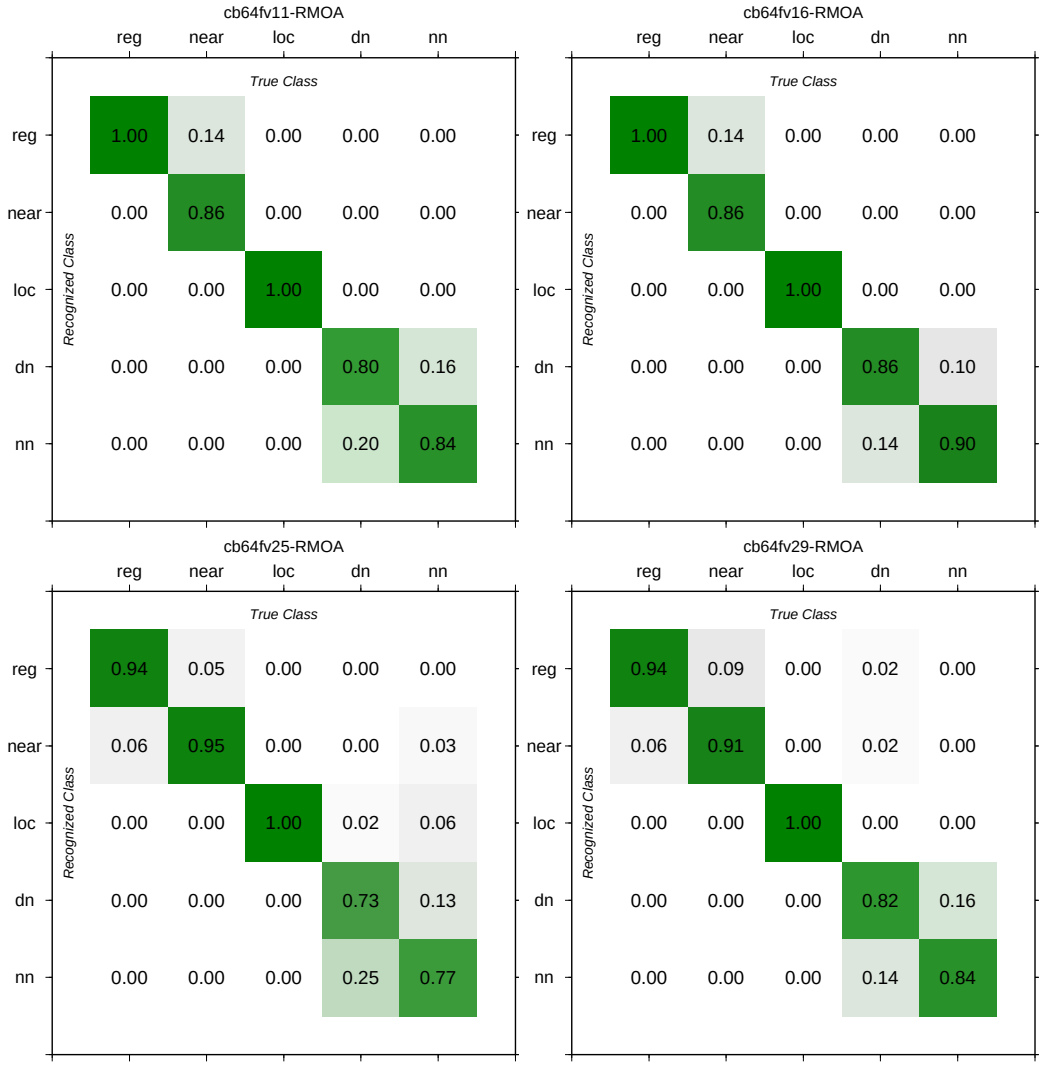

Figure 2.9: Confusion matrix in case of the reevaluation of the training data set. Annotations as follows: cb64fv11-RMOA denotes codebook (cb) with size 64, feature collection fv11 (see Tab. 2.2) and the station name is RMOA

and classification as the in total 142 events in the training data set (see Tab. 2.1).

The continuous, one step detection and classification is based on Eq. 2.6, i.e. no predetector is used, and the postprocessing steps described in Sec. 2.2.4 are applied.
To provide a better understanding Fig. 2.11 shows an example of a continuous classified seismogram. In the top row a seismogram containing two `loc` earthquakes is shown in black. The bottom row displays the corresponding averaged log probabilities for each class. The log probabilities are calculated in a sliding frame (frame rate 0.25) with class dependent length (Fig. 2.4) and averaged over three HMMs per class (Eq. 2.6). It can be seen that the probability of the `loc` class increases during the local events. At the same time the probability of the two noise classes decreases (the onset of the probability

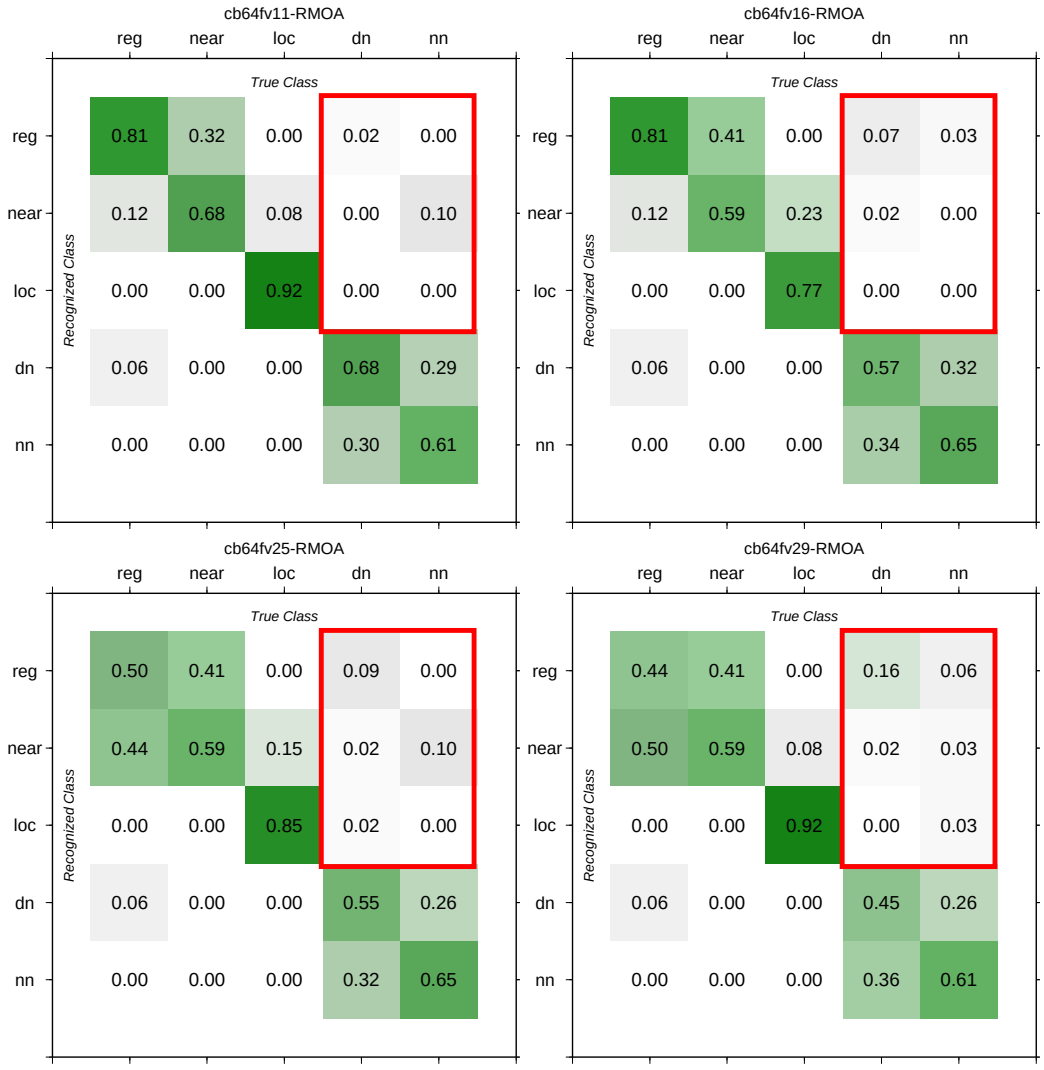

Figure 2.10: Confusion matrix in case of the "leave on out" method. Annotations as follows: cb64fv11-RMOA denotes codebook (cb) with size 64, feature collection fv11 (see Tab. 2.2) and finally the station RMOA. The position of the false alarms in the confusion matrix are emphasized by a red rectangle.

change lies actual before the `loc` events due to the 3s window of the feature generation). A classification is raised as soon as the log probability of an earthquake class is greater than the probability of a noise class and the seismogram is colored according to the presently active earthquake class. The coda of the `loc` class is confused as `near`, however, in the postprocessing the minimum event length, introduced in Sec. 2.3.2, tries to avoid confused events in that case.

The performance was tested for a one month period starting on June 1st 2004. Certainly earthquakes falling in this time period are excluded from the training data set. For the evaluation of the results a recursive STA/LTA, which proved to be suf-

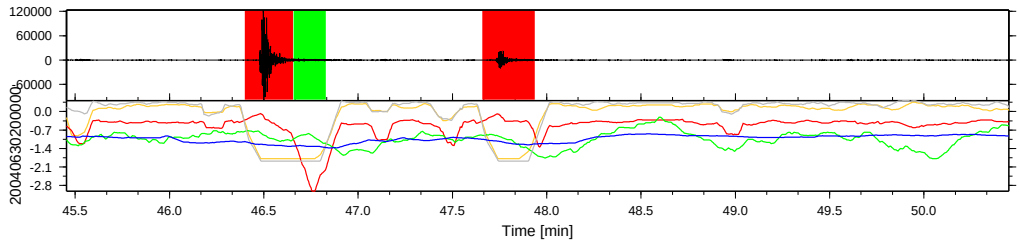

Figure 2.11: Top: Seismogram of two local events recorded with vertical component. Color sections correspond to final continuous classification without post-processing. Bottom: Log probabilities for each class (blue ≡ `reg`, green ≡ `near`, red ≡ `loc`, yellow ≡ `dn` and gray ≡ `nn`).

ficient as event detector regarding its low computational costs and good performance (Withers et al., 1998), was applied to the same time period. As trigger parameters a 1s STA window, a 30s LTA window and threshold 4 respectively 1.5 for switching the trigger on and off were used (Trnkoczy, 1998). The coincidence of the three stations was calculated such that an simultaneous detection was raised if at least two of three stations detected an event. The results in the continuous case are shown in Tab. 2.3, the comparison to the recursive STA/LTA is shown in Tab. 2.4 and discussion follows in the next section.

DHMM	# of `loc`		# of `near`		# of `reg`	
missed events:	8	(22%)	0	(0%)	5	(26%)
confused events:	3	(78%)	6	(100%)	4	(74%)
correct events:	26		7		10	
total earthquakes:	37		13		19	

Table 2.3: Classification performance of DHMM on a one month continuous seismic period.

	DHMM		rec. STA/LTA	
missed events:	13	(19%)	7	(10%)
detected events:	56	(81%)	62	(90%)
total earthquakes:	69		69	

Table 2.4: Comparison of the detection rates (without classifying) between DHMM and the recursive STA/LTA on a one month continuous seismic period.

2.4 Discussion

In the continuous case one month, i.e. $30 \cdot 24 \cdot 60 \cdot 60\mathtt{s}/0.25\mathtt{s} = 10368000$ (frame rate 0.25s, see Sec. 2.2.4) symbol sequences, are selected from the continuous symbol stream and classified using DHMMs. The DHMMs detect (correct + confused events) 78% of all `loc` events, 100% of all `near` events and 74% of all `reg` events (see Tab. 2.3). Measuring the classification performance DHMMs classify 26 (70%) `loc` events, 7 (54%) `near` events and 10 (53%) `reg` events correctly.
The intrinsic similarities between the `reg` and `near` class are the origin of higher confused event rates and consequently lower classification rates; already apparent in the "leave one out" method (Fig. 2.10). Joining the classes `reg` and `near` to a `regnear` class results in 27 ($27/32 \approx 84\%$) correct events for the `regnear` class, together with the unchanged 27 (70%) correct events for the `loc` class.
In order to rate these results a recursive STA/LTA detector is applied to the same time period. The recursive STA/LTA detects 90% of all earthquakes (sum of `loc`, `near`, `far` events) as compared to the DHMMs with an overall detection rate (correct + confused events) of 81% (see Tab. 2.4). The false alarms of both methods are comparable with 223 for the DHMM and 219 for the recursive STA/LTA. However, in contrast to the recursive STA/LTA the DHMM not only detects but also classifies the detection. Furthermore, when comparing the DHMM detection and classification rates (Fig. 2.3) with other classification algorithms, the erroneous detection rate of the pre-detector (here 10%) still needs to be subtracted. This means e.g. that 80% successful classifications for a pre-triggered ANN classifier would reduce to respectively 72% (Langer et al., 2006; Ibs-von Seht, 2008).

Certainly an improvement of the DHMM classification could be achieved by running first the recursive STA/LTA detector and then the HMM on the pre-detected data, similar to Langer et al. (2006); Ibs-von Seht (2008). Especially the good performance in the "leave one out" method would suggest this approach.
However, the main goal of this study was to present a detection and classification system which needs not to rely on a prior detection and thus can operate directly on the continuous data, which consequently allows the classification of low SNR earthquakes which would usually fall through the standard detector.

Even with a small sized data set (in total 142 sequences), the DHMM detection and classification achieves reasonable results. In contrast to template based pattern recognition, where templates of the whole training data set need to be stored for later classification, for DHMMs one probabilistic model per class is estimated from the training data and in the following the training data can be discarded. This allows the use of nearly unlimited amounts of training data (e.g. 160000 sequences in speech recognition). By running the DHMM continuously, earthquakes will be detected and classified which then can sequentially be added to the training to estimate improved DHMMs.

Furthermore a problem we have not addressed so far is the occurrence of mixed earthquakes, i.e. two or more earthquakes whose signals are running into each other. Defi-

nitely there is need for more research in this area.
Another point is, as already mentioned, a potential loss of information due to vector quantization of the DHMM. The migration to continuous HMMs which incorporate multivariate probability densities could improve the results and be less sensible to the feature selection.

2.5 Conclusion

We presented an alternative method for detecting and classifying earthquakes. This method is not based upon the usual system which is a short time average/long time average (STA/LTA) detector combined with e.g. an Artificial Neuronal Network (ANN) or template based pattern recognition. We prefer Hidden Markov Models (HMM) because of their strength in single step detection and classification, their explicit incorporation of the time and their major success in speech recognition. In speech recognition HMMs are favored over Time Delay Neuronal Networks (TDNN), an implementation of time windows in several ANN layers in order to process time dependent signals (e.g. Waible et al., 1989), or over template based pattern recognition in combination with Dynamic Time Wrapping (DTW), a similarity measure of signals which may vary in time or speed (e.g. Myers et al., 1980). Both show lower performance. HMM as TDNN and template based pattern recognition are not based on the time and amplitude dependent seismogram itself but on features estimated from the seismogram which clearly characterize the different classes. Special emphasis in this study was given to the determination of these features which is crucial for the performance of the classifier.
For testing the performance of this technique, discrete HMMs (DHMMs) are applied to three stations of the Bavarian earthquake service. The signal or class separation in this case is done by simply using the epicentral distance of earthquakes with respect to the stations used. Therefore it is possible to easily decide whether a classification is correct or wrong. Using this possibly smaller data set enables us to better evaluate the advantages and disadvantages of the proposed algorithm and to compare the results with simple and widely used detection techniques. The application to this data set is also of interest as for further hypocenter determination on real time analysis it is crucial to use cost intensive non linear techniques for local earthquakes inhabiting strong 3D topographic site effects and classical techniques for nearby and regional earthquakes. Thus the class memberships need to be known in advance.

Overall, the performance of the discrete HMM (DHMM) shows good results for the pre-triggered “leave one out” method with no missed events at all and 4% false alarms. For the continuous case DHMMs are used to detect and classify a continuous one month period. In order to rate the DHMM results a recursive STA/LTA detector is applied to the same period. When comparing only the number of detected events, which underestimates the performance of the DHMM method as it not only detects but also classifies, the recursive STA/LTA detection performs slightly better with a

90% detection rate compared to the DHMM detection and classification with 82% respectively. However, when comparing the DHMM to other classification systems, where the erroneous detection rate of the pre-detector needs to be included, the DHMM detection and classification certainly shows comparable results.

The purpose of this paper was to demonstrate the possibility of a classification system which directly operates on continuous data, thus making it possible to detect low SNR earthquakes which usually would fall through the pre-detector. The possibility of applying such a one step detection and classification system increases in relevance as more and more stations continuously record data.

Based on the aspects discussed in this paper, one step HMM detection and classification is competitive to common detection and classification systems.

2.6 Acknowledgments

We want to thank Matthias Ohrnberger whose Ph.D thesis (Ohrnberger, 2001) and various discussions were the bases for this work. Also we want to thank Heiner Igel for his kind support and the staff of the Geophysical Observatory in Fürstenfeldbruck who maintains the stations used in this work.
The suggestions of two anonymous reviewers substantially improved the manuscript.

3 Continuous Hidden Markov Models: Application to Automatic Earthquake Detection and Classification at Las Cañadas Caldera, Tenerife

by Moritz Beyreuther, Roberto Carniel and Joachim Wassermann

Published in Journal of Volcanology and Geothermal Research, 2008, 176(4), 513-518

Abstract

A possible interaction of (volcano-) tectonic earthquakes with the continuous seismic noise recorded in the volcanic island of Tenerife was recently suggested. Also recently the zone close to Las Cañadas caldera shows unusual high number of near (<25km), possibly volcano-tectonic, earthquakes indicating signs of reawakening of the volcano putting high pressure on the risk analyst. Certainly for both tasks consistent earthquake catalogues provide valuable information and thus there is a strong demand for automatic detection and classification methodologies generating such catalogues. Therefore we adopt methodologies of speech recognition where statistical models, called Hidden Markov Models (HMMs), are widely used for spotting words in continuous audio data. In this study HMMs are used to detect and classify volcano-tectonic and / or tectonic earthquakes in continuous seismic data. Further the HMM detection and classification is evaluated and discussed for a one month period of continuous seismic data at a single seismic station. Being a stochastic process, HMMs provide the possibility to add a confidence measure to each classification made, basically evaluating how "sure" the algorithm is when classifying a certain earthquake. Moreover, this provides helpful information for the seismological analyst when cataloguing earthquakes. Combined with the confidence measure the HMM detection and classification can provide precise enough earthquake statistics, both for further evidence on the interaction between seismic noise and (volcano-) tectonic earthquakes as well as for incorporation in an automatic early warning system.

3.1 Introduction

The basaltic shield structure of Tenerife (Canary Islands, Spain), built between 12 and 3.3 Ma, is now hosting the 3.5 Ma Las Cañadas volcanic complex (Martí et al., 1994), including a caldera due to a series of collapses (Martí and Gudmundsson, 2000). The most recent structure within the caldera is the Teide-Pico Viejo volcanic complex (TPV), 200 Ka, that produced both basaltic effusive (e.g. 1909 Eruption of Chinyero) and phonolitic explosive eruptions (e.g. Montaña Blanca, about 2 Ka). This bimagmatism poses serious difficulties on hazard assessment.
Several seismic stations have been recently installed to improve monitoring after the appearance of anomalous clusters of seismic events below the island. These stations have been recording not only discrete events but also a continuous seismic noise, which shows a strong anthropogenic component but definitely contains a significant natural signal. A review of the analysis carried out on this seismic noise can be found e.g. in Carniel et al. (2008a).
An important issue is the possible influence of regional (i.e. also purely tectonic, non-volcano-tectonic) events on the time features of the seismic noise, an influence that has been noted elsewhere before (e.g. Ambrym, see Carniel et al., 2003; Stromboli, see Carniel and Tárraga, 2006). Evidence exists of significant eruptions being triggered in the past by regional earthquakes, so that the study of this influence can be extremely important for hazard assessment (e.g. Tungurahua, see Tárraga et al., 2007; Villarrica, see Tárraga et al., 2008; Raoul Island, see Jolly and Scott, 2007). Manga and Brodsky (2006) present an extensive review of other cases. It has been demonstrated that even very distant earthquakes can have an influence on active calderas (e.g. Nisyros, see Gottsmann et al., 2007).

In the specific case of Tenerife, a bi-directional interaction between the time evolution of the continuous seismic noise and the occurrence of (volcano-) tectonic events is postulated. On one hand continuous seismic noise evolution can be used to forecast the occurrence of volcano-tectonic events at least in a statistical sense (Tárraga et al., 2006), on the other hand tectonic events are associated with changes in the intensity, spectral and/or dynamic features of continuous seismic noise (Carniel et al., 2008a). The statistics that one can build regarding these possible interactions is however significantly reduced by the low signal to noise ratio (SNR) that usually characterizes the seismic data recorded on the island, and only few attempts have been made in order to detect events with algorithms more sophisticated than the classical STA/LTA ratio (see e.g. Del Pin et al., 2008). There is therefore a strong demand for automatic detection / classification methodologies able to process considerable amounts of data in a reasonable computing time and build a consistent statistics of tectonic events occurrence times, that can be then compared with time series of changes visually observed (Carniel et al., 2008a) or statistically detected (Fattori Speranza and Carniel, 2008) in the seismic noise.

In this paper we take the approach to automatic detection / classification of volcano-tectonic and tectonic events on Tenerife island by adopting speech recognition method-

ologies. One of the motivation for doing so is based on the intrinsic similarity between seismogram signals and speech signals (see Fig. 3.1 or Ohrnberger (2001)). In speech recognition Hidden Markov Models (HMMs) are successfully applied for spotting spoken words in continuous audio data. Therefore we propose to use HMMs for spotting earthquakes in continuous seismic data, eliminating the need for any pre-triggering and thus being predestined for low SNR.

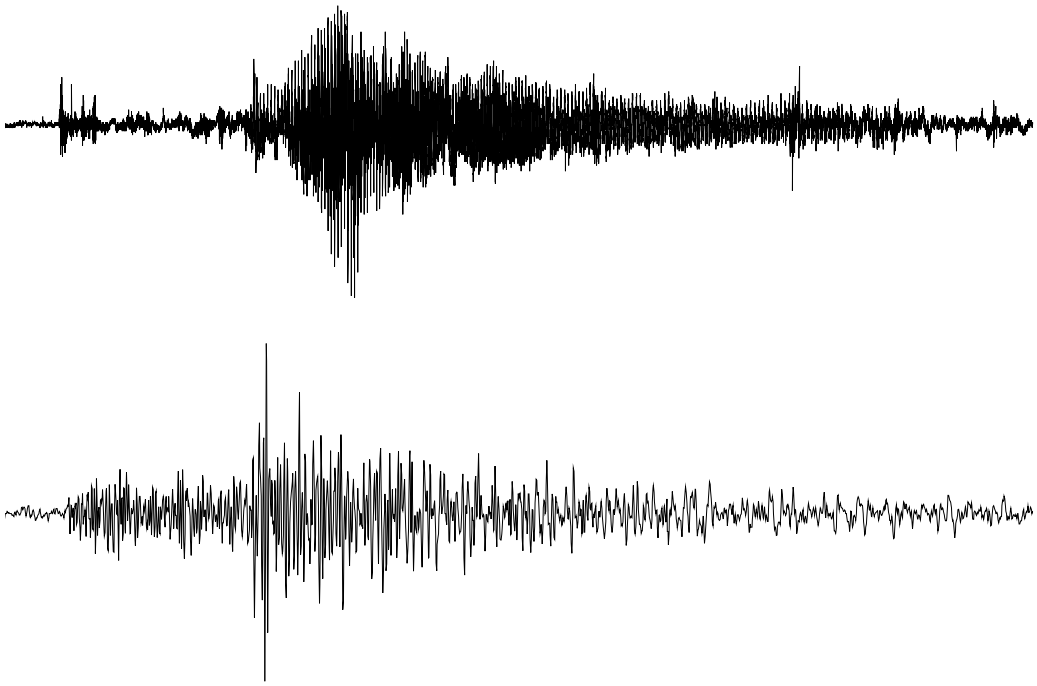

Figure 3.1: Top figure shows an audio signal of the word "dry" compared to the bottom figure showing an earthquake recorded at BDG station. The word "dry" has a duration of 2.7s, the earthquake has a length of 37s. The similarity between vocal utterances and earthquakes is one of the motivations to use speech recognition techniques for earthquake detection and classification.

An informal, illustrative insight into the structure of HMMs is given in the next section together with a probabilistic treatment of the detection and classification task. A roadmap of applying HMMs to automatic detection and classification at BDG station (Tenerife) is following. Further the performance on a continuous one month period is evaluated, discussed and conclusions are drawn.

3.2 Theory of Continuous Hidden Markov Models

Before providing a probabilistic treatment of the detection and classification task, some informal insight into the structure of HMMs is given. Fig. 3.2 illustrates an example HMM generating a seismogram. The model is assumed to be composed out of the three states (1, 2, 3). The observation $\mathbf{o}_t = o_t$ is considered to be the amplitude of the seismogram at time t, thus the seismogram of length T can be represented as a sequence of observations, i.e. amplitude values, $o_1 o_2 \cdots o_T$.
The model changes state i every time t, where the probability for changing to state j is given by the state transition probability a_{ij}. The state transitions to state j are assumed to be independent of all previous states except the most recent i. Generally the allowed state transitions are drawn as arcs (arrows) in the model, thus in the HMM of Fig. 3.2 only state transitions into the same state or into the next state are allowed. Each time t a state i is entered an observation is generated from the observation probability distribution $b_i(o_t)$, which differs for each state i. E.g. in Fig. 3.2 the observation probability distribution for the first state $b_1(o_t)$ gives high probability for generating low to medium amplitude values; it does not forbid to generate high amplitude values, it just makes them less probable. This built-in flexibility makes the HMM very powerful.

The HMM can be seen in Fig. 3.2 from a generative viewpoint as follows: The observed amplitude values of the first part of the seismogram are obtained by first choosing to stay in (changing into) state 1 for $t \in [1, t_1[$. Each time $t \in [1, t_1[$ staying in state 1 the observed amplitude value o_t is sampled from the first state observation probability distribution. The amplitude values of the middle part of the seismogram are then sampled from the second state observation probability distribution $b_2(o_t)$ with $t \in [t_1, t_2[$. The probability distribution $b_2(o_t)$ in Fig. 3.2 gives higher probability for high amplitude values, thus resulting in higher sampled amplitudes; low amplitude values are still not impossible, just less probable. The final part of the seismogram is then sampled from $b_3(o_t)$ with $t \in [t_2, T]$ conditioned on staying in state 3.
Thus the probability for the observation sequence, i.e. the seismogram, conditioned on the model of Fig. 3.2 can be calculated by multiplying the temporal evolution of the state transition probabilities and the corresponding observation probabilities of the amplitude values o_t:

$$\begin{aligned} P(o_1 o_2 \cdots o_T | \{a_{ij}, b_i(); \forall i, j\}) = 1 \cdot b_1(o_1) a_{11} b_1(o_2) a_{11} b_1(o_3) \cdots a_{12} b_2(o_{t_1}) a_{22} \cdot \\ \cdot b_2(o_{t_1+1}) \cdots a_{23} b_3(o_{t_2}) a_{33} b_3(o_{t_2+1}) \cdots a_{33} b_3(o_T). \end{aligned} \tag{3.1}$$

The parameters $b_i(), a_{ij}$ are organized in an overall structure called HMM $\boldsymbol{\lambda} = \{a_{ij}, b_i(); \forall i, j\}$. The 1 in Eq. 3.1 accounts for the initial state probability of the first state. This probability is 1 due to the restricted model in Fig. 3.2 which allows no alternative initial state.

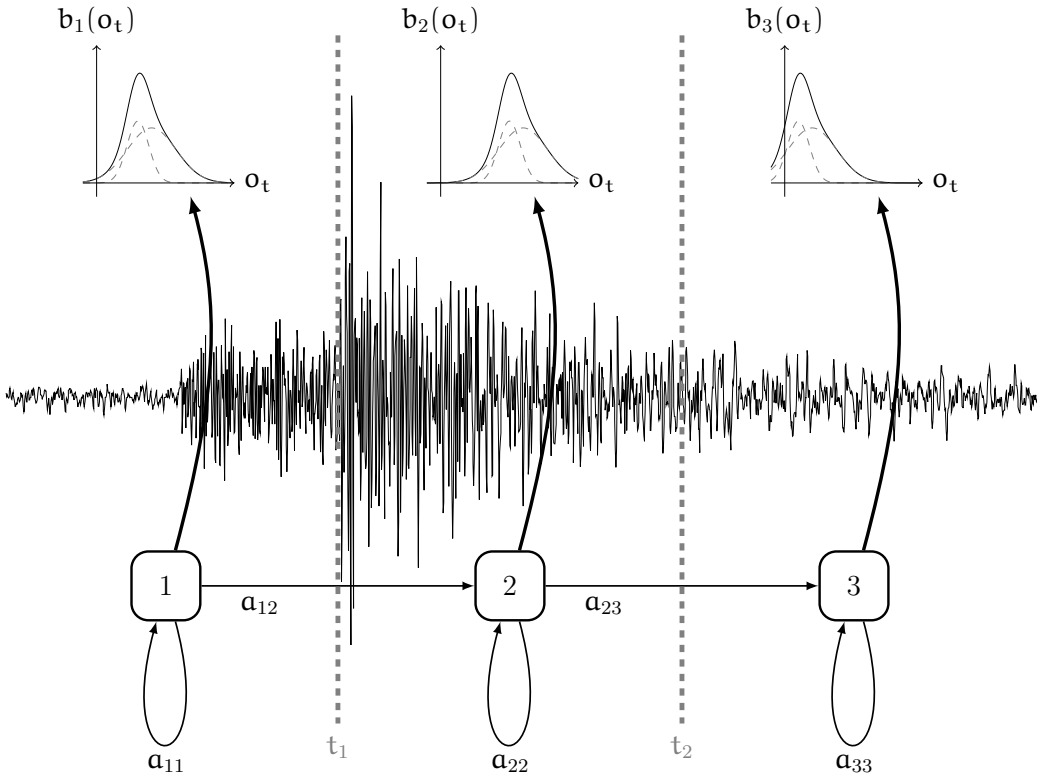

Figure 3.2: Sketch of a HMM generating a seismogram: The observed amplitude of the first part of the seismogram is generated from the observation probability distribution $b_1(o_t)$, i.e. the model stays in (changes into) state $q_t = 1$ for every sample $t \in [1, t_1[$. The middle part of the seismogram is controlled by $b_2(o_t)$, $t \in [t_1, t_2[$, thus resulting in higher amplitudes. And the final part of the seismogram is generated from $b_3(o_t)$, $t \in [t_2, T[$.

The probability of some event X, given the occurrence / instantiation of Y is known as conditional probability and denoted by $P(X|Y)$. Thus in the framework of conditional probabilities the model parameters can also be defined as follows:

- The state transition from state i to state j is the probability that the next state q_{t+1} is j given the occurrence of the current state q_t is i: $a_{ij} = P(q_{t+1} = j|q_t = i)$.
- The observation probability distribution $b_i(\mathbf{o}_t)$ of state i is the probability of the observation $\mathbf{o}_t$ conditioned that the current state q_t is i: $b_i(\mathbf{o}_t) = P(\mathbf{o}_t|q_t = i)$.

By providing a model like the one shown in Fig. 3.2 it is possible to calculate the probability of the observation sequence $P(o_1 o_2 \cdots o_T|\boldsymbol{\lambda})$ given the HMM $\boldsymbol{\lambda}$.

The task in the classification is now to find the most probable class k, given the observation sequence $\mathbf{O}$. By effectively replacing the class k by the class representative

HMM $\boldsymbol{\lambda}_k$, the classification task can be formulated as the most probable model $\boldsymbol{\lambda}_k$, given the observation sequence (seismogram) $\mathbf{O} = \mathbf{o}_1\mathbf{o}_2\cdots\mathbf{o}_T$:

$$k = \arg\max_k P(k|\mathbf{O}) = \arg\max_k P(\boldsymbol{\lambda}_k|\mathbf{O}). \tag{3.2}$$

The probability of the model given the observation $P(\boldsymbol{\lambda}_k|\mathbf{O})$, likewise the HMM $\boldsymbol{\lambda}_k$ which are trained based on class dependent training data sets, are calculated using well established algorithms of speech recognition (for general information on HMM see Rabiner, 1989; Young et al., 2002; Schukat-Talamazzini, 1995).

Although depicted in Fig. 3.2 for illustration purposes, the amplitude is of course not a relevant observation, also called feature, in which earthquake classes differ and thus can easily be distinguished. Other features like the spectrogram, the envelope or the instantaneous bandwidth are ways better suited. In general a mixtures of multivariate Gaussian distribution $b_i(\mathbf{o}_t)$ is used to estimate a multi dimensional feature vector $\mathbf{o}_t$. Having given a short informal insight into the theory of HMM, the realization of a HMM based earthquake detection and classification algorithm is described in the next section.

3.3 Realization of a HMM based Earthquake Detection and Classification Scheme on Tenerife

On the island of Tenerife, four major seismogram classes are identified (compare to Fig. 3.3): The "NEAR" class, which corresponds to earthquakes inside the 25km boundary (possibly volcano-tectonic); the "FAR" class corresponding to earthquakes outside the 25km boundary but inside a 75km distance (mainly tectonic). This boundary was proposed for discrimination of volcano-tectonic and regional tectonic earthquakes by Tárraga et al. (2006). The "DN" class, accounting for noise during day (more anthropogenic) and the "NN" class accounting for noise during night (more non-anthropogenic). Example seismograms for all classes recorded at BDG station are shown in Fig. 3.4. BDG is a short period seismic station installed on the island of Tenerife (see Fig. 1 in Del Pin et al., 2008, for details on the location) by the Departamento de Volcanología, CSIC, Madrid (Ortiz et al., 2001); BDG uses a 1Hz Mark sensor and continues sampling with 16bit at 50Hz.

Different features were calculated from the seismogram and the following were chosen to be relevant for the detection and classification task at BDG station: The short time Fourier transform in 10 binned logarithmic frequency bands ranging from 0.5Hz - 25Hz. Joswig (1994) shows that this features, also called sonogram, gives distinct information of the particular seismogram. A normalized version of the instantaneous amplitude, also known as seismic envelope (Beyreuther and Wassermann, 2008a), the time derivative of the envelope called instantaneous bandwidth (Barnes, 1993) (i.e. a

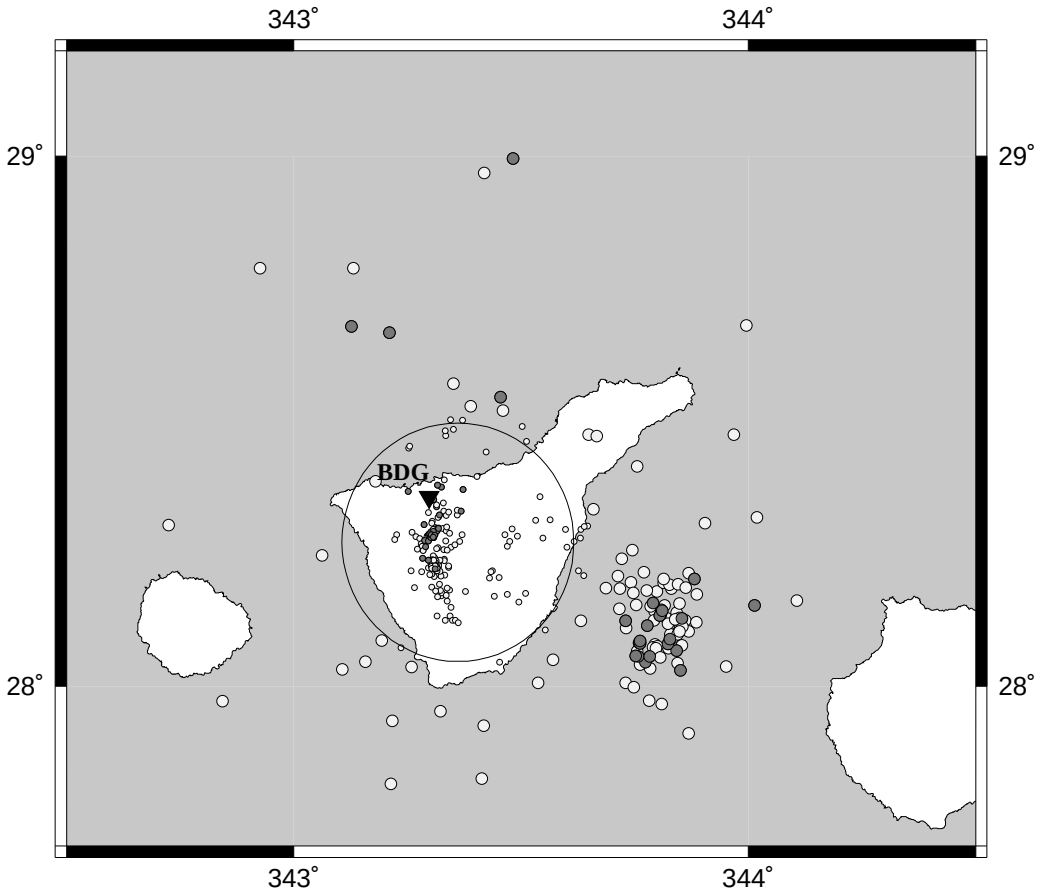

Figure 3.3: Map showing 25km region around Teide volcano. Earthquakes falling inside the region are represented as small circles in light gray, earthquake outside the 25km region are represented as medium circles. Small circles falling outside the 25km region are due to 2 digit accuracy of the IGN catalogue. Circles in dark gray depict earthquakes in the training data set.

delta impulse has unlimited bandwidth), the instantaneous frequency (Taner et al., 1979) and the centroid time (Ohrnberger, 2001) which is the time in a sliding window where half of the area below the envelope is reached. Thus, all together 14 features are used, resulting in a 14 dimensional feature space. The features are generated in a tapered 3s window with step size 0.04s. Full description of the calculation of the features can be found in Ohrnberger (2001); Beyreuther and Wassermann (2008a).

For each class training samples are selected based on classifications of the IGN catalogue (http://www.ign.es, see also Fig. 3.3). In total 119 samples for the DN class, 65 samples for the NN class, 25 samples for the NEAR class and 23 samples for the FAR class were selected. Example seismograms of the training data sets are shown in Fig. 3.4. Due to the low SNR we were only able to extract 23 FAR earthquakes which were visible for the seismological analyst at BDG for the training (large dark circles in Fig. 3.3).

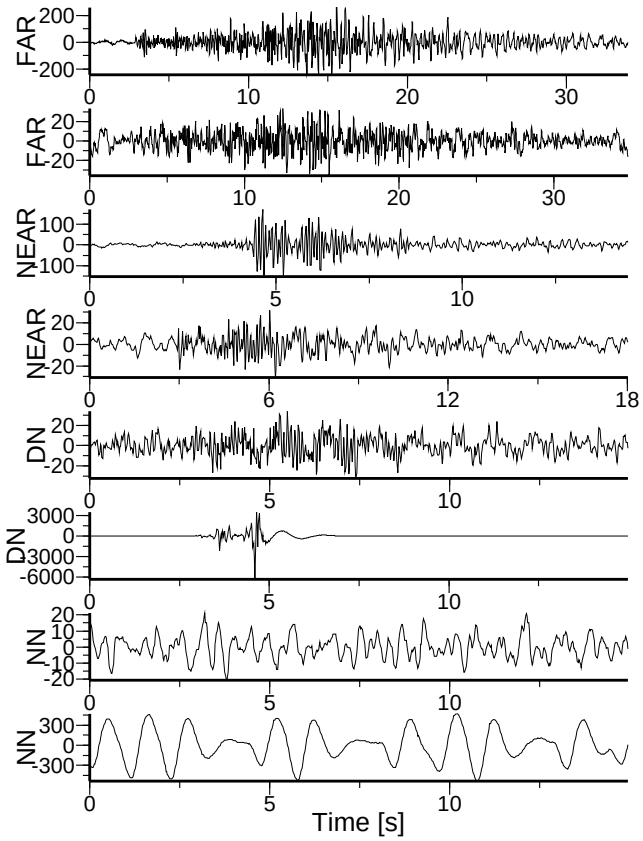

Figure 3.4: Typical seismograms for the classes FAR, NEAR, DN and NN. Far earthquakes have origin outside the 25km zone (see Fig. 3.3), the origin of the NEAR earthquakes is inside the 25km zone, the DN class stands for mostly anthropogenic noise during day time and the NN noise class depicts mostly non-anthropogenic noise during night.

Even though there were more training data for the NEAR class available we used only a subset (small dark circles in Fig. 3.3) in order to approximately match the size of the FAR training data. This training samples are then used to compute the HMMs with the HTK software (Young et al., 2002).

In order to realize an "earthquake spotter", crude noise models (DN and NN), known as garbage, filler or catch-all models (e.g. see Knill and Young, 1994), are trained to catch all "non-earthquakes". This is achieved by training the variances of the garbage models, i.e. the variance of $b_i(o_t)$, together with earthquake models (FAR and NEAR). The result is a vague catch-all model since it has wide variance and therefore poor probabilistic scores. Consequently the score of the garbage models (DN and NN) is tuned relative to the earthquake models (see left plot in Fig. 3.5).
Next to the vague garbage models good models are needed for the earthquake classes.

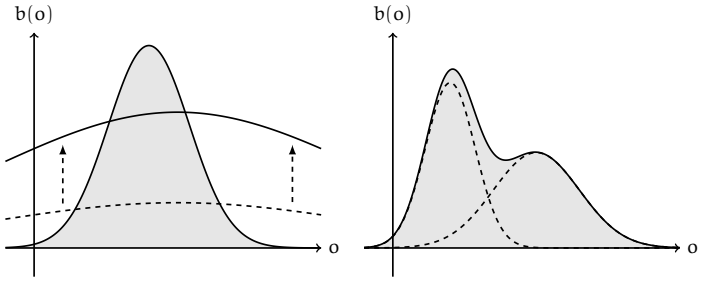

Figure 3.5: The left plot shows in gray the probability distribution of the (FAR or NEAR) earthquake model relative to probability distribution of the (DN or NN) garbage model denoted with a dotted line. Since the garbage model has wide variance and therefore poor probabilistic scores, the garbage models (DN and NN) are tuned relative to the FAR and NEAR class. The probability distribution of the tuned garbage model is shown by the shifted solid line. The plot on the right shows two Gaussian mixtures, denoted by dotted lines, whose superposition allows a better fit to an arbitrary probability density; thus provides a good model for the earthquake classes. The superposition, shown as solid line, is filled with gray color.

Because the training data size is quite low compared to speech recognition with training data bases consisting of 160000 telephone utterances (Qing et al., 2000), there is a tradeoff between finding a good model and having limited amount of data. In this study we resolve this tradeoff by splitting the variance of the earthquake classes (NEAR and FAR) into two mixtures in order to allow a better fit to the real observation probability distribution (see right plot in Fig. 3.5). At the same time each of the two mixtures are trained for the earthquake classes for all states together and shared. Sharing the variances results in robustness by effectively reducing the number of parameters to estimate. However the means are trained separately for each class.

Respecting the manipulations described in the last paragraph, HMMs are trained for each class based on the training data and are in the following used to detect and classify earthquakes. As soon as the HMMs are trained for each class the training data can be discharged resulting in a fast classification by Eq. 3.2. Due to taking the argument of the maximum in Eq. 3.2 all the probability information of the resulting classification is lost. This can be circumvented by also calculating how many times the probability of detecting an earthquake exceeds the probability of detecting noise. This probability measure, called confidence (known as likelihood scores in Knill and Young, 1994, p. 3-5) is computed by dividing the probability of detecting an earthquake class by the probability of detecting a noise class:

$$\text{conf.} = \frac{\mathsf{P}(\boldsymbol{\lambda}_{k\in\{\mathsf{FAR},\mathsf{NEAR}\}}|\mathbf{O})}{\mathsf{P}(\boldsymbol{\lambda}_{k\in\{\mathsf{DN},\mathsf{NN}\}}|\mathbf{O})}. \tag{3.3}$$

E.g. a confidence level of 2 means that the algorithm computes two times higher probability of detecting an earthquake than of detecting noise. In the implementation

two classifiers are running in parallel, one containing all classes and one with only DN and NN classes. The probabilities for the corresponding HMMs are calculated in a 16s sliding window with 50% overlap.

The classifier is evaluated in a continuous one month period starting at 1. September 2004 and ending at 30. September 2004. Earthquakes in this period are excluded from the HMM training data sets. In one month $30 \cdot 24 \cdot 3600\mathrm{s}/(16\mathrm{s} \cdot 50\%) = 324000$ observation sequences are evaluated using the HMM detector and classifier. 324000 is further used as the total number of noise events, neglecting the small number of earthquakes, in order not to clutter Tab. 3.1. The results are given for the whole test data set and a small subset containing only events with SNR higher than 2. The noise amplitude is measured as the maximum amplitude in a one minute time window before the event. The signal amplitude is measured as the maximum amplitude in a window containing the signal.
An independent classification was performed by two experienced seismological analysts on the same time period as follows: in case a clear earthquake was found which was not included in the IGN catalogue, it was added to the earthquake list. In case a vague signal was visible in the time frame where IGN classified an earthquake the earthquake was added to the earthquake list. On the other hand vague signals which were not in the IGN catalogue were rated as false alarms. Thus the resulting correct detections are a lower bound of the real correct detections. The results are shown in Tab. 3.1 and further discussed in the next section.

3.4 Discussion

The final results are shown in Tab. 3.1. There are roughly $139/30 \approx 4.6$ false alarms, $26/30 \approx 0.9$ correct classified earthquakes and $8/30 \approx 0.3$ missed earthquakes per day. The false alarms with 4.6 per day are still a reasonable amount to verify for the seismologist. At the same time at a lower bound 76% of the earthquakes are correctly detected, 24% are missed. IGN catalogued 35 earthquakes in this period, 27 of them were visible at BDG, 8 were not present — not event as vague signal. On the other hand 7 earthquakes which were not catalogued by IGN were detected and classified.
Certainly for an observatory mainly focused on cataloguing earthquakes the number of missed earthquakes are not really acceptable and multi station networks or higher quality data are to be taken into account. But for an observatory mainly focussed on monitoring the alert level of volcanoes, the number of false negatives is acceptable and the HMM detection and classification is a promising starting point.

Due to the probabilistic framework it is possible to calculate the confidence (Eq. 3.3) of the correct classified earthquakes. Tab. 3.1 shows that the algorithm is quite "sure" as soon as it detects an earthquake; the probability of detecting an earthquake is on average 452.9 times higher than the probability of detecting noise. On the other hand, the algorithm is "unsure" when detecting a false alarm where the probability of detecting an earthquake is only on average 1.2 times higher than the probability of

	# of Events	Av. Conf.	Conf.>2 and SNR>2
missed earthquakes	8		3
correct earthquakes	26	452.9	18
	$26/34 \approx 76\%$		$18/21 \approx 86\%$
correct noise	323861		597
false alarms	139	1.2	3
	$139/324000 \approx 0.04\%$		$3/600 \approx 0.5\%$

Table 3.1: Classification results for a continuous one month period starting at 1. September 2004 and ending at 30. September 2004.

noise. Thus the confidence measure provides valuable information for the seismological analyst in order to evaluate the detections, where false alarms are mixed up with correctly classified earthquakes.

The low SNR ratio at BDG station combined with the good performance of the confidence measure suggests to evaluate the performance on a small subset of the data set with high SNR. Therefore we choose only earthquakes having SNR higher than 2, resulting in 21 earthquakes. At the same time the detections were restricted to solely confidence greater than 2, i.e. we require the algorithm to give a twice as high probability of detecting an earthquake than of detecting noise, resulting also in 21 detections. 3 earthquakes were missed and there are also 3 false alarms. These 18 highly confident earthquakes contain 86% of the major, i.e. high SNR, earthquakes recorded at BDG station. Therefore 18 of the 26 correct classified earthquakes could be marked as highly confident earthquake detections providing valuable information for the seismological analyst. On the other hand $34-21 = 13$ earthquakes ($13/34 \approx 35\%$) have SNR lower than 2. The algorithm is still able to detect $26 - 18 = 8$ low SNR earthquakes ($8/13 \approx 62\%$), which are certainly tough to extract using the standard detectors, (e.g. STA/LTA, Withers et al., 1998).

So far only the detection results were discussed. But the big advantage of using HMM is that they are designed to detect and classify signals of multiple different class identities in one step, easily fast enough to run online. The earthquakes in the one month continuous classification test split up in 3 FAR earthquakes and 30 NEAR earthquakes. While the algorithm classified all 3 FAR earthquakes, i.e. 100%, correctly as members of the FAR class, 4 NEAR earthquakes, i.e. $4/30 \approx 13\%$, were misclassified as member of the FAR class. Certainly the number of tested FAR earthquakes is too small to draw any conclusions but the high number of correct classifications of NEAR earthquakes, 87%, seem to be a good foundation for an automatic continuous classification.

Before providing a conclusion we want to show how this work compares to previous studies of HMM applied to seismic signals on the one hand and provide some background on the differences between the "memory" of the seismic signals on Tenerife and "memory" of the HMM on the other hand:

I) In this paper we used continuous HMM, implemented and developed by the HTK (Young et al., 2002) software package. In comparison to previous applications, where discrete HMM were used (Ohrnberger, 2001; Beyreuther and Wassermann, 2008a), continuous HMM are directly used to estimate the time evolution of multi dimensional feature vectors $\mathbf{O}_t = \mathbf{o}_1 \cdots \mathbf{o}_T$ and thus makes the discretization step and the resulting loss of information of discrete HMM unnecessary. The dimensionality combined with the limited amount of training data made it then necessary to manipulate the variance in the training in order to obtain robust models (see previous section and Fig. 3.5).

II) The "memory" of HMM constrained by the state transition probabilities $a_{ij} = P(q_{t+1} = j|q_t = i)$ given in the theory section. That is with a 0.04s step size of the feature generator the next state j is depending only on the current state i (0.04 seconds before, not taking into account the window size of 3s for feature generation and the restrictions due to the allowed transitions, the arrows in Fig. 3.2). However Carniel et al. (2008b) show with the variogram method that the memory of the seismic noise at Teide-Pico Viejo is roughly 70 hours. This apparent contradiction is easily resolved by the fact that in this study we train good HMM for the earthquakes themself and use rough catch-all HMM for the noise classes, making it unnecessary to precisely model the 70 hours memory of the (seismic) noise. It is also worthwhile to note that the training of models with 70 hours memory would be computational infeasible.

3.5 Conclusions

The detection and classification of one month continuous raw seismic data recorded at BDG station during the period 1. September 2004 — 30. September 2004 highlighted the performance of HMM applied to detection and classification of seismic signals of possible volcanic origin. Even in low SNR, where it is definitely hard for a seismological analyst to detect and classify earthquakes, HMM combined with the confidence measure can provide helpful information.
Seismic catalogues, like IGN, which are based on multi station networks are truly irreplaceable. However, a significant part of the earthquakes catalogued in IGN are not visible at BDG. This is due to the low SNR resulting from the the strong attenuation of the volcanic rock (Gudmundsson et al., 2004) and a general high noise level near the station. On the other hand small earthquakes near BDG station are, due to the strong attenuation, not detected at the seismic network on which IGN is based. Therefore the possibility of the HMM method for running automatically on a local station can provide better information on earthquakes recorded at that single station. This information about earthquakes measured, together with the analysis of seismic noise, is precise enough to provide more statistical evidence supporting the hypothesis of a relationship between the seismic noise and the occurrence of earthquakes (see Tárraga et al., 2006; Carniel et al., 2008a).
The precision of the HMM detection and classification on BDG data, restricted to this single station, is not yet enough for automatic cataloguing of earthquakes (but

certainly promising enough to call for further applications on multi station or higher quality data). However certainly HMM can be used in an automatic online alert level estimation system, where especially the 86% correct high confident (conf. > 2) earthquakes detections restricted to SNR greater than 2 together should prove to be valuable; the low false alarm rate of 0.5% is a necessary criterion in such a scenario.
In summary, single station HMM detection and classification can automatically provide precise enough earthquakes statistics recorded at BDG station which then can be used for further investigation of the Teide-Pico Viejo volcanic system and, especially equipped with the confidence information, provide helpful information for the seismological analyst who catalogues earthquakes and the volcanic risk analyst.

3.6 Acknowledgements

This work was carried out in the framework of the Alpe Adria Research Grants (Italy), the PPP / VIGONI grant (DAAD D/05/54450, CRUI 436-2006) supporting the exchange between Friulian and Bavarian scientists.
We authors wish to thank Heiner Igel for his kind support. The Madrid CSIC-MNCN research group provided the BDG seismic data in the framework of the projects TEIDEVS (Spain, CGL2004-05744) and "Azioni Integrate Italia-Spagna MIUR-MEC 2007" . The seismology and functional mechanics research groups in Udine are acknowledged for useful discussions. Finally, we also wish to thank the authors of the HMM Tool Kit (HTK) (Young et al., 2002) for providing and developing such a powerful software package.
The suggestions of Olivier Jaquet and of an anonymous reviewer substantially improved the readability of the manuscript.

4 Hidden semi-Markov Model based earthquake classification system using Weighted Finite-State Transducers

by Moritz Beyreuther and Joachim Wassermann

Published in Nonlinear Processes in Geophysics, 2011, 18, 81-89

Abstract

Automatic earthquake detection and classification is required for efficient analysis of large seismic datasets. Such techniques are particularly important now because access to measures of ground motion is nearly unlimited and the target waveforms (earthquakes) are often hard to detect and classify. Here, we propose to use models from speech synthesis which extend the double stochastic models from speech recognition by integrating a more realistic duration of the target waveforms. The method, which has general applicability, is applied to earthquake detection and classification. First, we generate characteristic functions from the time-series. The Hidden semi-Markov Models are estimated from the characteristic functions and Weighted Finite-State Transducers are constructed for the classification. We test our scheme on one month of continuous seismic data, which corresponds to 382629 classifications, showing that incorporating the time dependency explicitly in the models significantly improves the results compared to Hidden Markov Models.

4.1 Introduction

The automatic detection and classification of seismic signals is increasing in significance since data centers have moved from the acquisition and archiving of single data snippets to streaming continuous seismic waveforms. Automatic detection and classification of earthquakes is used, for example, to automatically acquire consistent earthquake catalogues at volcanoes, to achieve class-dependent pre-selection of localization methods,

and to exclude quarry blasts from earthquake catalogues.
Our choice to reach a robust detection and classification algorithm is to adopt Hidden Markov Models (HMMs). This technique is very successfully applied to speech recognition (Young et al., 2002), and has practical applications to the field of seismology (Ohrnberger, 2001; Beyreuther and Wassermann, 2008a) and other fields (Kehagias and Fortin, 2006). The advantages and disadvantages of this technique compared to other approaches is thoroughly covered in Beyreuther and Wassermann (2008a) for the field of seismology.
HMMs provide a powerful tool to describe highly variable time series based on double stochastic models. The model acts on characteristic functions or features (estimated from the seismogram), e.g. the envelope or the power in different frequency bands, which describe the earthquake better than the pure ground motion signal itself. One part of the stochastic model represents the time dependency of these derived characteristic functions; the other part represents the distribution of the characteristic functions itself. Since this is a fully probabilistic approach, a confidence measure is naturally also provided.
However, a drawback when using HMMs is that the probability of the duration for a single part in the HMM (called state) is an exponentially decaying function in time which is an unrealistic representation for the duration of earthquake classes or speech (Oura et al., 2008). To overcome this limitation, we apply Hidden semi-Markov Models (HSMMs) which use the more realistic Gaussians as state duration probability distributions.

The commonly used HMM decoding/classification technique (Viterbi algorithm) cannot be applied to HSMMs, as it relies strongly on the intrinsic HMM design. Therefore, we construct Weighted Finite-State Transducers (WFSTs) from the HSMMs for the purpose of classification (Mohri et al., 2002). This step also allows a much more flexible model refinement in the actual classification.

In the next section the HMMs are briefly introduced to the field of seismology, followed by a more detailed description of their HSMM extension and the corresponding WFST classifier. In order to show the potential for earthquake detection and classification, we apply HSMM as well as HMM to a one month continuous seismic dataset.

4.2 Theory

4.2.1 Hidden Markov Models

Hidden Markov Models (HMMs) are estimated from a training data set, i.e. they belong to the class of supervised classification techniques. The models are not operating on the seismic signal itself but on characteristic functions (also called features) generated from the seismograms which better represent the different classes of earthquakes.

Figure 4.1a shows an example of a training data set in the time-amplitude space. The label corresponds to the class name where the classes simply differ in epicentral distance (`reg` 100km – 600km, `near` 10km – 100km, `loc` 0km – 10km, `noise` noise). Note that the `near` and `reg` class have a quite similar signature in the time-amplitude space and therefore are not easy to distinguish. A much better characterization of the classes is shown in Fig. 4.1b, where different characteristic functions are plotted. Each band corresponds to a plus minus one standard deviation band around the mean of the characteristic function amplitude over all available training data for that specific class. In this representation it is much easier to distinguish the different classes, as can be seen by comparing the `reg` and `near` class (blue and green line) in the characteristic function space and in the amplitude space, respectively.

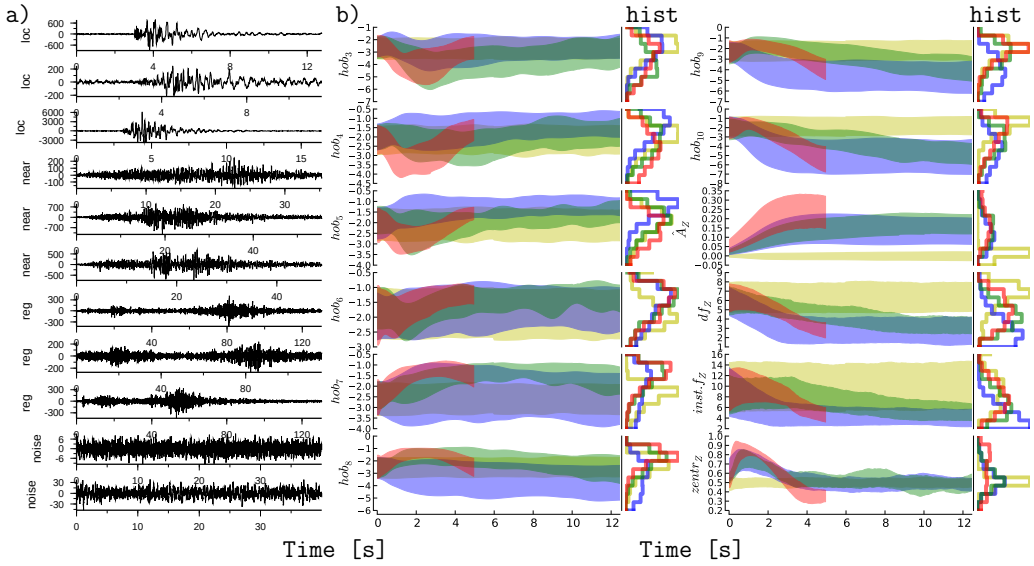

Figure 4.1: Training data examples a) and characteristic functions b) generated from the training data set (plus/minus one standard deviation around the mean). Color-coding: `reg` in blue, `near` in green, `loc` in red, `noise` in yellow. A histogram over the timespan is appended vertically for each characteristic function in b). 3s were subtracted from the start and added to the end of each sample in a) to emphasize the contrast to the noise.

Figure 4.1b also makes clear why it is important to include time dependency in the model. If time dependency is excluded, the classes can only be distinguished through their histogram (as appended vertically for each characteristic function). As an example the half octave band 10 ($\texttt{hob}_{10}$) may be taken: The `reg` and `near` class (blue and green line) are not easily distinguished in the time independent histogram (Fig. 4.1b). However they are easily distinguishable in the time dependent characteristic function itself. For more details on the use of characteristic functions as well as the selection process, see Ohrnberger (2001) or Beyreuther and Wassermann (2008a).

Figure 4.2 shows a sketch of a HMM for a single earthquake class. The observation

is usually a vector containing multiple features, though for visualization purposes we use a single sample point ($\mathbf{o} = \mathtt{o}$) which represents the absolute amplitude. The HMM segments the earthquake signal in different parts over time $\mathtt{i}$ (called states). For each part an observation probability distribution $\mathtt{b_i(o)}$ of the characteristic function (here the absolute amplitude) is estimated. The sequence of the different parts is controlled by the (state) transition probability ($\mathtt{a_{ij}}$), from part ($\mathtt{i}$) to part ($\mathtt{j}$). The probability of the observation sequence $\mathbf{o}_1, \mathbf{o}_2, \mathbf{o}_3, \mathbf{o}_4, \mathbf{o}_5$... can then be calculated by taking the product of observation probabilities and state transition probability for each observation sample, e.g.

$$\mathtt{b}_1(\mathbf{o}_1)\mathtt{a}_{11}\mathtt{b}_1(\mathbf{o}_2)\mathtt{a}_{12}\mathtt{b}_2(\mathbf{o}_3)\mathtt{a}_{23}\mathtt{b}_3(\mathbf{o}_4)\mathtt{a}_{32}\mathtt{b}_2(\mathbf{o}_5)..., \tag{4.1}$$

assuming that the state transition sequence is known/given. However, as the transition sequence is usually not known, the transition sequence which has the maximum probability is taken. Searching all possible transition sequences is incredibly time consuming and therefore usually an optimized algorithm (called Viterbi algorithm, see Rabiner, 1989) is used for classification.
In the HMM training procedure, the transition probabilities ($\mathtt{a_{ij}}$) and the observation distributions ($\mathtt{b_i}$) are estimated automatically through an expectation-maximization algorithm. For more details on the theory of HMMs, see Rabiner (1989) and Young et al. (2002).

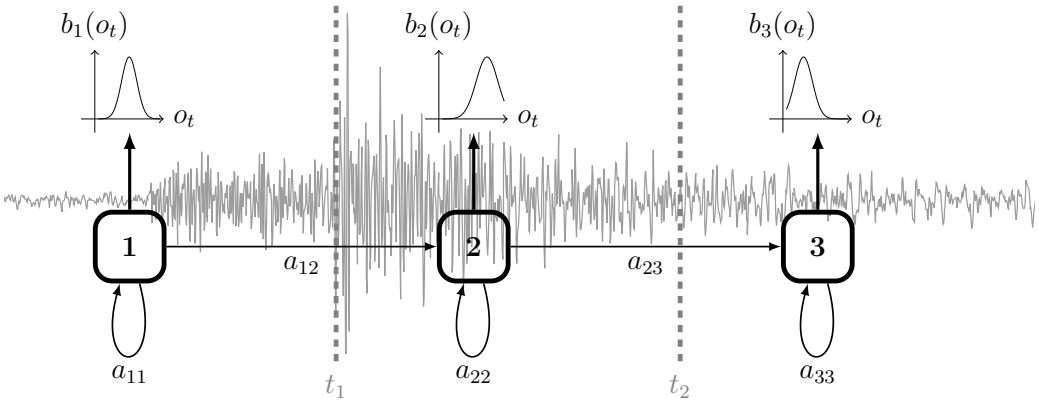

Figure 4.2: Hidden Markov Model with three different states. The state transition probabilities are denoted with $\mathtt{a_{ij}}$. The observation distribution for each state is denoted with $\mathtt{b_i(o_t)}$.

4.2.2 The Extension to Hidden semi-Markov Models

The duration probability distribution of a particular HMM state is not included in the intrinsic HMM design. Nonetheless, according to the definition of HMMs the

probability of staying T time steps in state i can be calculated from the state transition probabilities (a_{ij}) as follows:

$$a_{ii}^{T}(1 - a_{ii}) = (1 - a_{ii}) \exp(T \log(a_{ii})) = const \cdot \exp(-T|\log(a_{ii})|); \tag{4.2}$$

with a_{ii} being the self-state transition probability ($0 \leqslant a_{ii} \leqslant 1$), which yields the negative logarithm in the third part of the equation. The result is an exponentially decaying function of the duration time T. An exponentially decaying function, however, is not an adequate representation of the duration distribution of certain states (e.g., P-wave and P-coda, S-wave and S-coda etc.) of an earthquake class, as this would imply that this part (state) has most likely length one or zero (see Fig. 4.3).

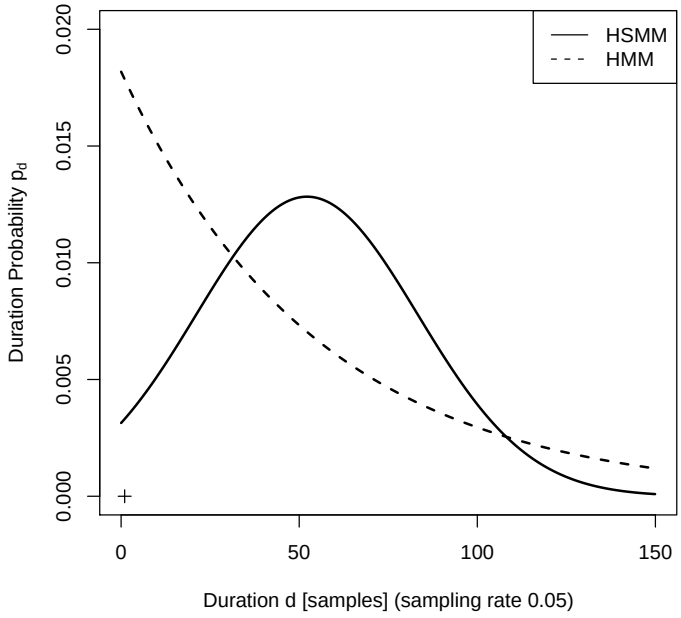

Figure 4.3: Exponential (HMM) versus Gaussian (HSMM) state duration probability distribution.

An alternative representation is to integrate the state duration probability distributions explicitly into the HMM. This HMM extension is known as a Hidden semi-Markov Model (HSMM) (Oura et al., 2006, 2008). In doing so, we are now able to approximate the duration probabilities through Gaussians. The HSMMs were a breakthrough in speech synthesis because it is crucial that certain speech parts have the correct duration since they otherwise sound unnatural (Zen et al., 2004).
Figure 4.4 provides an example of a three state Hidden semi-Markov Model. First a sample d_1 is drawn from the duration probability distribution (Gaussian) of the first state δ_1. Depending on the value of d_1, d_1 observations $o_1 ... o_{d_1}$ are generated with the corresponding probabilities $b_1(o_1) ... b_1(o_{d_1})$. Then the second state is entered and the same procedure is continued for all remaining states. The corresponding probabilities

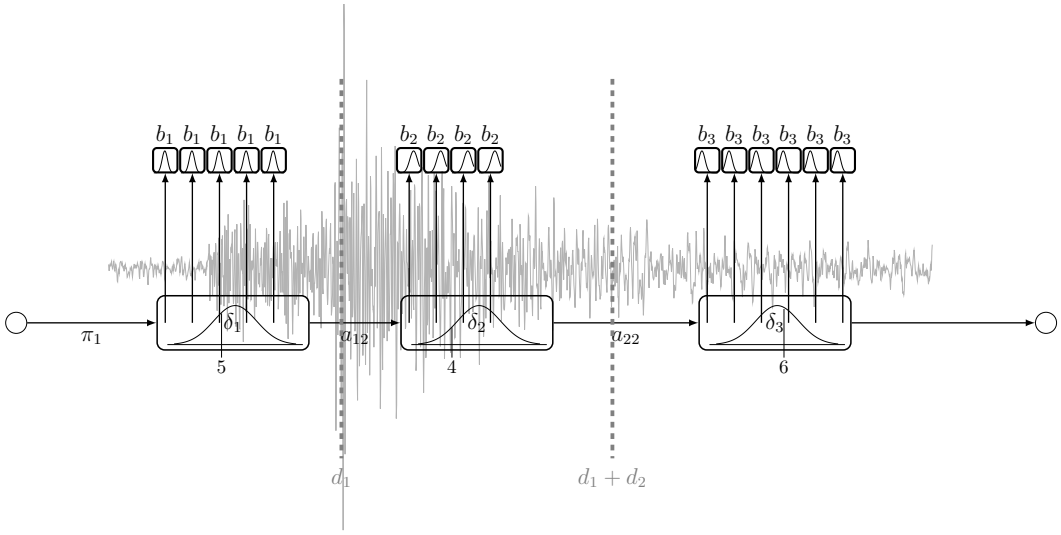

Figure 4.4: Hidden semi-Markov Model. The duration probabilities $\delta_{1,2,3}$ are plotted in the states (lower rectangles). Directly when a state is entered, a sample is drawn from δ, e.g. five in the part. Consequently five observations are generated from the observation probability b_1 and then the next state is entered.

are multiplied:

$$\delta_1(d_1) \prod_{i_1=1}^{d_1} b_1(\mathbf{o}_{i_1}) \quad \delta_2(d_2) \prod_{i_2=d_1}^{d_1+d_2} b_2(\mathbf{o}_{i_2}) \quad \delta_3(d_3) \prod_{i_3=d_1+d_2}^{d_1+d_2+d_3} b_3(\mathbf{o}_{i_3}); \tag{4.3}$$

with $d_1 + d_2 + d_3 = T$. The probability of the single HSMM given an observation sequence $\mathbf{o}_1 ... \mathbf{o}_T$ is the one with the values of d_1, d_2, d_3 which maximize Eq. 4.3. Again the observation distributions (b_i) and the duration probability distribution (δ_i) are estimated automatically during HSMM training (for details see Oura et al., 2006).

4.2.3 Weighted Finite-State Transducer decoding

The Viterbi algorithm is standard in HMMs classification (also called decoding). However, this cannot be applied for HSMMs because its optimized dynamic programming core relies strictly on the architecture of the HMM state transitions. Viterbi decoding is optimized and therefore extremely fast. In contrast, Weighted Finite-State Transducers (WFSTs) unfold the complete HSMM structure, where each unfolded path is modeled individually at the expense of speed. However, by associating weights such as probabilities, durations, or penalties to each possible path, the WFSTs provide a unified framework for representing and refining various models (used by Mohri et al., 2002, for speech and language processing). WFSTs are the standard HSMM decoder

used in speech recognition (see Oura et al., 2006) and in following we show how to build up an HSMM decoder with a WFST.

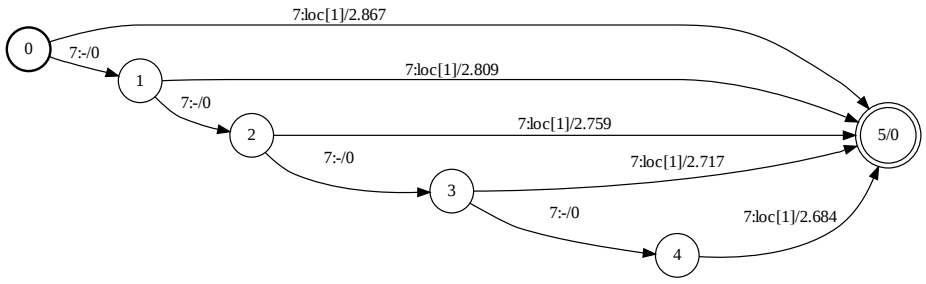

Figure 4.5: The Weighted Finite-State Transducer for the first 5 time samples in the `loc` model. The lowest path (with 5 arcs) corresponds to the first part in Fig. 4.4 which is bound to the right by the dashed line labeled $\mathtt{d}_1$. While the nodes only mark the steps in time, the arcs contain the input states, output labels, and the corresponding weight, which in this case is the negative log probability of the duration probability.

HSMM state	reg 1	reg 2	reg 3	near 1	near 2	near 3	loc 1	loc 2	loc 3	noise 1
WFST input label	1	2	3	4	5	6	7	8	9	10

Table 4.1: The mapping of HSMM states and WFST input labels. Each earthquake class has three states.

Figure 4.5 shows the first five possible paths for the first state of the `loc` model. In total, the number of paths per state in our application ranges between 24-875; however, for visualization purposes, we show the first five only. The five observations in Fig. 4.4 (first "loc" state) are represented by the lowest path in Fig. 4.5. The other paths (Fig. 4.4), with 4,3,2 and 1 arcs, would correspond to the drawn values 4,3,2 and 1 of δ_1. The circles in Fig. 4.5 correspond to numbered nodes which the transducer can pass. All the probabilities are stored in the arcs and described by the corresponding labels. The annotation "7:-/0" for instance corresponds to the WFST input label 7 (the first of loc, see Tab. 4.1), no output label ("-") is assigned, and the negative log probability for this transition is 0 which corresponds to $\mathtt{exp}(-0) = 100\%$. The annotation "7:loc[1]/8.032" assigns the output label "loc[1]" to the path (in principle any name can serve as a label; we chose the name of the HSMM state as output label) and a negative log probability of this transition of 8.032 which corresponds to $\mathtt{exp}(-8.032) = 0.033\%$. Therefore, all we need to do to build an earthquake classifier for the first `loc` state with a WFST is to link the input label to an HSMM state (Mohri

et al., 2002) and to assign the negative log probability of the last arcs in the transducer to the duration probability of all previous arcs, thus setting their probability to 100%. The probability for staying two samples in state one (the second path from the top in Fig. 4.5) corresponds to $\texttt{cdf}(2) - \texttt{cdf}(1)$, whereby the cumulative duration distribution function is denoted as `cdf`.

The WFSTs allow composition, unification and concatenation of the transducers (for more details, see Mohri et al., 1997). In order to build up a WFST for the complete `loc` earthquake we simply concatenate the transducer for the first, second, and third `loc` state into one overall `loc` earthquake transducer. Figure 4.5 shows the transducer for the first `loc` state and Fig. 4.6 represents the concatenated transducer for the whole `loc` model.

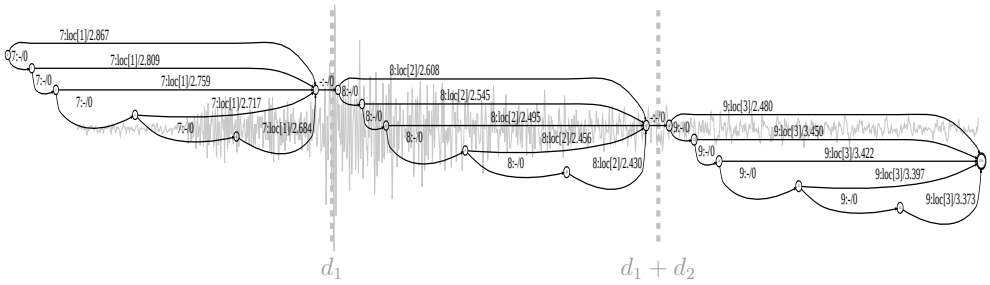

Figure 4.6: Concatenated Weighted Finite-State Transducer. The WFST is the result of concatenating the WFST shown in Fig. 4.5 for the first, second and third `loc` state. The labeling of the above figure is described in Fig. 4.5. The corresponding mapping from HSMM states to WFST input labels is described in Tab. 4.1. Note that for visualization, only the first 5 paths per HSMM state are plotted.

The classifier for the reevaluation of the training data is constructed by unifying the concatenated WFSTs of the earthquake classes and the noise as shown in Fig. 4.7. In the classification process the negative log-likelihoods are minimized to find the most likely path. The WFST for the reevaluation of the training data in the next sections has in total 10^{16} different possibilities/paths.
The flexibility in constructing the WFST easily allows the introduction of minimum and maximum length criteria for the earthquake parts. By, for example, deleting the arcs pointing from nodes $(0 \rightarrow 5)$ and $(1 \rightarrow 5)$ in Fig. 4.5, the resulting WFST has a minimum duration of two samples. In the classification of an earthquake it should be impossible to directly travel from HSMM state $(1 \rightarrow 2 \rightarrow 3)$ and stay during all the remaining samples (order of 100 or 1000) in the third HSMM state, which then basically resembles only the third HSMM state. In order to avoid this behavior, and to increase the speed of the application, we build the WFST with the minimum length being the value of the 30^{th} percentile and the maximum length being the 70^{th} percentile of the duration distribution for each state respectively.

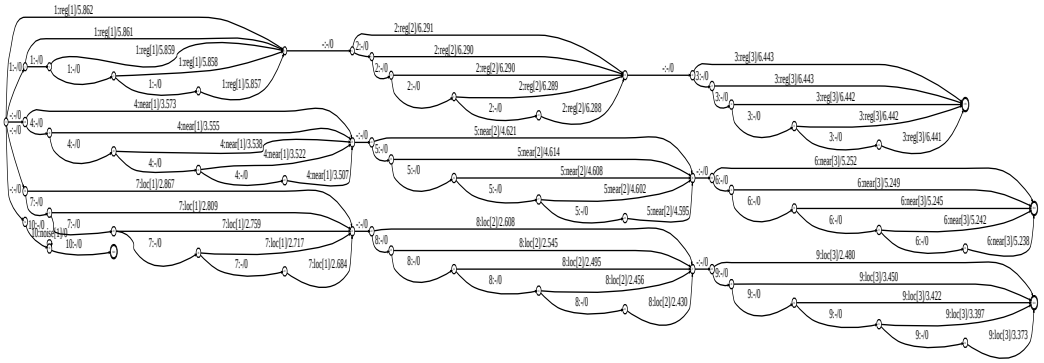

Figure 4.7: Classifier design for the reevaluation of the training data. By concatenating the different HSMM states, one WFST per earthquake class is constructed (Fig. 4.6). The separate WFST for each class can then be unified (i.e. put in parallel) which results in one large WFST shown here. Note again that for visualization, only the first 5 paths per HSMM state are plotted.

4.3 Application

The HSMM detection and classification system was applied to a one month period (2007-09) of continuous data from the seismic station RJOB of the Bavarian Earthquake Service (http://www.erdbeben-in-bayern.de). The specified data set was chosen because the target classes `loc`, `near` and `reg`, which served as examples in the theory section and are represented and differ by epicentral distance. This enables us to easily decide whether a classification is correct or not, thus allowing a better evaluation of the advantages and disadvantages of the proposed algorithms. The training data set consists of 122 `loc`, 37 `near`, 115 `reg` and 176 `noise` events.

A first performance test is the reevaluation of the training data set. The models are trained from the training data and then as a second step, the same training data are then reclassified by using these trained models (allowing only one class per event as shown in Fig. 4.7). The results are shown in Fig. 4.8. The detection performance of HMM versus HSMM (Fig. 4.8) is expressed by the rate of false alarms and missed events. The missed event rate is higher for HSMM, whereas the false alarm rate is higher for the HMM. A higher false alarm rate is problematic, because during continuous classification mostly noise is classified (e.g. 370151 noise classifications in the following continuous period). Thus, even a low percentage of false alarms lead to a high total number. Nonetheless, the classification performance of both HMM and HSMM is similar. The reevaluation of the isolated training data set provides an indication of the performance, since over-training and performance in various noise conditions is not covered. Therefore we also chose to classify one month of continuous data.

In one month 370151 classifications are made ($3600 \cdot 24 \cdot 30/7$, with 7s window step size). The continuous data are classified in a sliding window of class dependent length (`loc`

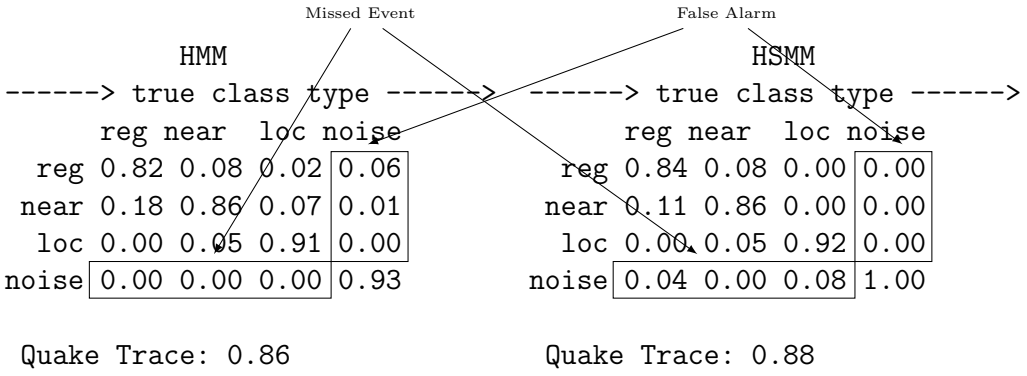

Figure 4.8: Reevaluation of the training data. To the right the true class type is plotted; to the bottom the recognized one. E.g., for the HMM `reg` class, 82% of the data are correctly classified as `reg` and 18% are confused as `near`.

14s, `reg` 157s, `near`: 47s), with a window step of 7s. The window length was chosen such that each class fits about twice inside the window. For each window and type, a [noise, earthquake, noise] and an only [noise] sequence are classified. This allows a flexible position of the earthquake in the window and calculation of the confidence measure P(*earthquake*)/P(*noise*), with P being the probability.

Figure 4.9 displays the results of the HMM and HSMM classification. The plot DATA shows three local earthquakes; the plots HSMM and HMM show the results of the two different models, respectively. The log probabilities of the earthquake models are plotted as colored bars, and the log probabilities of the corresponding noise model are plotted as dashed lines in the same color. An earthquake is detected if the probability for the earthquake is greater than the probability for noise. The corresponding type is classified by selecting the earthquake class with the highest confidence measure, which in Fig. 4.9 corresponds to the vertical distance of the associated noise probability to the center of the earthquake probability (displayed as a bar). Other classifications in that period are deleted, even if they are longer than the class with the highest confidence measure. This procedure is known as "the winner takes all" (see e.g. Ohrnberger, 2001). By comparing the results from the HMM and HSMM classification, it is easy to see that the classes are much better characterized by the HSMM. Through their rather simple design the HMM even missclassifies the second and third local earthquake. It is also clear to see that the `reg` HSMMs have more realistic minimum duration (the 10s duration of the `reg` HMM for the third local earthquake is impossible) due to the more flexible model design in the WFST.

This classification procedure is applied to a one month period of continuous data. In order to avoid a large number of false alarms, minimum confidence thresholds for a classification are introduced (for details see Beyreuther et al., 2008b), which are chosen in such a way that the false alarm rate for both HMM and HSMM are similar. The results are shown in Tab. 4.2.

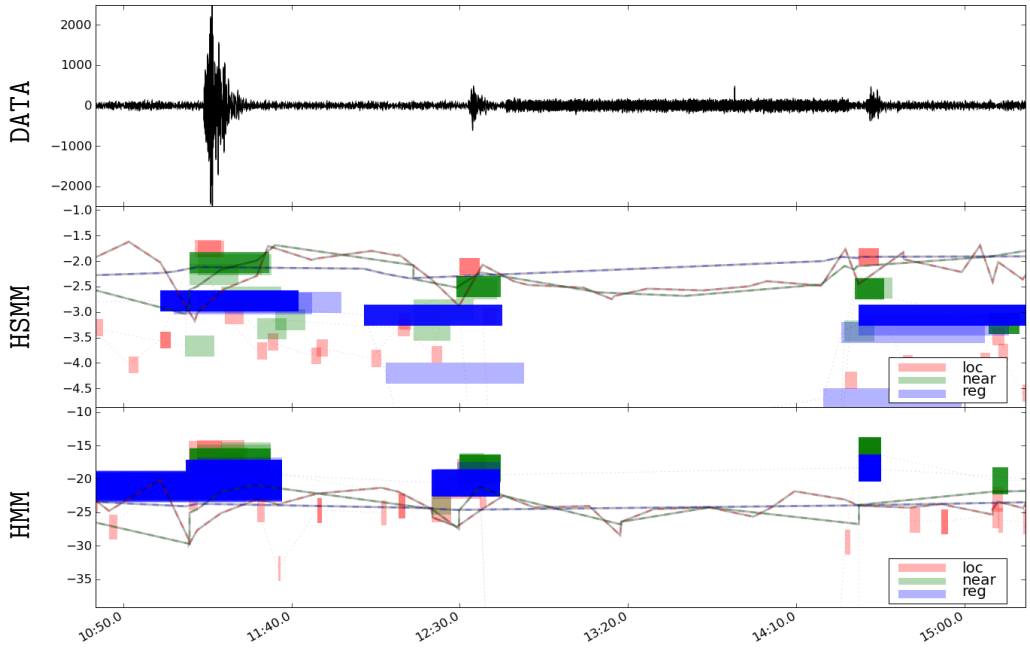

Figure 4.9: The top plot shows three local earthquakes. The second and third plot shows the resulting log probabilities from HMM and HSMM detection and classification respectively. Colored bars denote the log probability of the earthquake classes, while the dashed lines denote the corresponding noise classes for each earthquake class.

The HSMM clearly outperforms the HMM. However, the results of the HMM and HSMM could be easily improved by using a different parameterization, e.g. more states. But since the computation time of the WFST decoding would then increase dramatically, we choose to use a rather simple setup. The reasons for choosing certain parameters and the parameters itself are discussed in the following.

4.4 Discussion

A key point for the HMM and HSMM design is the refinement to the correct parameter set. The parameters are the number of Gaussians which compose the observation probabilities (called Gaussian mixtures), the number of states in general, the model topology and the possibility of tying variances etc. The motivation for choosing the parameters and the parameters themselves are explained in the following.

1.) We use left to right models as explained in the theory section due to the causality of the earthquakes (Fig. 4.1). For noise, which is clearly time independent (Fig. 4.1), only one state is used and thus no time dependency is assumed. Due to the limited

HMM	# of `reg`		# of `near`		# of `loc`		# of `noise`
correct events:	9	(26%)	4	(25%)	56	(67%)	370084
confused events:	5	(14%)	1	(6%)	12	(14%)	67
missed events:	21	(60%)	11	(69%)	15	(18%)	–
total:	35		16		83		370151

HSMM	# of `reg`		# of `near`		# of `loc`		# of `noise`
correct events:	23	(66%)	10	(63%)	75	(90%)	370087
confused events:	5	(14%)	2	(13%)	5	(6%)	64
missed events:	7	(20%)	4	(25%)	3	(4%)	–
total:	35		16		83		370151

Table 4.2: Results of one month of continuous classification. Confused events are events which are misclassified to another earthquake class. The confused events for the `noise` class correspond to the false alarms. For the comparison, the post-processing thresholds for both HSMM and HMM are chosen such that the false alarm rate stays for both algorithms at about two false alarms per day.

amount of training data, it is statistically likely to over-train the model parameters. Consequently for a robust estimation of the parameters, only a limited amount of states and Gaussian mixtures for the observation probability $\mathtt{b}_\mathtt{i}(\mathbf{o})$ can be used. Also, the high computational costs of the WFST classification limits us to a maximum amount of six states (0.5 hours CPU time for 1h of data).

2.) Because of the time-dependent characteristics of the earthquakes themselves (Fig. 4.1), we use more discretizations in time (6 states) for the earthquake models and only one for the noise model.

3.) For the noise model we choose a higher number of Gaussian mixtures (4 mixtures) for the observation probability distributions $\mathtt{b}_\mathtt{i}(\mathbf{o})$ such that a large amount of different noise sources could be matched. For the earthquake models we use a single mixture.

4.) In previous studies day noise and night noise classes were used, which match the anthropogenic noise during day time and the non-anthropogenic noise during night time (Ohrnberger, 2001; Beyreuther and Wassermann, 2008a). In this study, however, we use only one noise class in order to avoid competing day noise and night noise models through the competitive training of Young et al. (2002); Zen et al. (2009).

5.) The training of Young et al. (2002) and Zen et al. (2009) allows the computation of full multidimensional covariance matrices for the observation probability. However, in order to keep the number of free parameters low, the characteristic functions are first transformed to their principal axes (using principal component analysis), such that only the diagonal of the covariance matrices need to be used.

We also tried other parameter combinations: earthquake states $\mathtt{i}$ from three to six, Gaussian mixtures of $\mathtt{b_i(o)}$ from one to six, and tied variances for all models. Clearly a lower number of earthquake states results in a lower number of discretizations in time and thus a worse classification result. More interestingly, a higher number of Gaussian mixtures for the observation probability distributions of the earthquake models did not achieve better results. This may indicate that the characteristic functions of the earthquakes are well represented by one Gaussian distribution. Also a higher amount of mixtures for the noise model did not achieve better results; four mixtures seem to be robust.

4.5 Conclusions

Seismology is a data rich science which allows nearly unlimited access to measures of ground motion. The migration of data centers' data acquisition from event based data snippets to continuous data has increased the amount of available waveforms dramatically. However, the target waveforms (earthquakes) are hard to detect, particularly with low signal to noise ratios. Advanced detection and classification algorithms are required, e.g. for automatically acquiring consistent earthquake catalogues for volcanoes, for class-dependent pre-selection of localization methods, or for excluding explosions from earthquake catalogues.

In this study we applied Hidden Markov Models, which are double stochastic models known from speech recognition for classification. We demonstrated the benefit of including a more realistic time dependence in the model (HSMM). The classification of one month of raw continuous seismic data (in total 370151 classifications) shows a classification performance increase up to 40% using HSMM vs. HMM.
However the performance increase has to be balanced with the increase of CPU time by a factor of 10 (1/2h CPU time for 1h data), thus making it difficult to process large data archives. For future work either a much more optimized WFST implementation needs to be used or the HMM models need to be refined in such a way that the natural duration of the earthquake parts is better represented.

This paper shows the improved performance from including more realistic time dependencies specifically for HMM. However, an increased performance should also be possible by including the time dependency in the model design of other supervised learning techniques such as support vector machines (Langer et al., 2009) or artificial neuronal networks (Ibs-von Seht, 2008) when classifying time series.
Another major impact on the detection and classification performance, especially the false alarm rate, is easily achieved by combining the results of several stations, similar to the coincidence sums of common triggers.

In order to compare the results of this study to other methods in earthquake detection and classification (e.g. artificial neuronal networks, support vector machines, self

organizing maps or tree based classifiers) benchmark data sets are required. Unfortunately these are currently not available for the field of seismology and, concluding the presented study, we think there is a strong demand for them.

4.6 Acknowledgements

The authors wish to thank Heiner Igel for his kind support. We also wish to thank the authors of the HMM Tool Kit (HTK), Young et al. (2002), the authors of the HMM-based Speech Synthesis System (HTS), Zen et al. (2009) and the authors of the AT&T FSM Library™, Mohri et al. (1997) for providing and developing such powerful software packages.
The suggestions of three referees, Wu Ye and two anonymous reviewers, substantially improved the manuscript.

5 Design of a Probabilistic Earthquake Detector

Moritz Beyreuther, Conny Hammer, Joachim Wassermann, Matthias Ohrnberger and Tobias Megies

Submitted to Geophysical Journal International, 09/2010

Abstract

In the case that only one or two seismic stations are available, common triggers often show too high false alarm rates and thus are hardly applicable. As an alternative, we design a single station probabilistic earthquake detector with performance rates similar to common triggers with additional coincidence over several stations. The probabilistic architecture is adapted from double stochastic models called Hidden Markov Models which are widely used in speech recognition. However, many pitfalls need to be circumvented in order to apply speech recognition technology to earthquake detection. Throughout the paper we concentrate on the adaption steps and show many examples. The Hidden Markov Models are derived from simple Gaussian classifiers. We apply the methodology to the detection of anthropogenic induced earthquakes and demonstrate for a 3.9 months continuous seismic period that the single station probabilistic earthquake detector achieves similar detection rates as a common trigger in combination with coincidence sums over three stations.

5.1 Introduction

For detecting earthquakes, triggers are the common method of choice (Withers et al., 1998). Especially when combining their output from multiple stations, the detection rates are high and the false alarm rates low. However, if there are only one or two stations are available, the false alarm rate increases rapidly and, in the end, the seismologist usually has to interactively re-analyse the continuous data.

In this study we design a probabilistic earthquake detector which shows good performance when applied to a single station and thus is extremely valuable in scenarios where only one or two stations are active. The probabilistic architecture has been

adopted from speech recognition, the corresponding double stochastic model is known as Hidden Markov Model (HMM). In geoscience, HMMs are mainly known from the field of volcano seismology, where they show good performance in classifying seismic data of volcanic origin (Ohrnberger, 2001; Beyreuther et al., 2008b; Ibáñez et al., 2009). To demonstrate their potential for seismology, Fig. 5.1 shows 10h of seismic data recorded at Mt. Merapi including the color coded results of the HMM classification. Fig. 5.1 emphasizes the large potential of the HMMs: Even with a high number

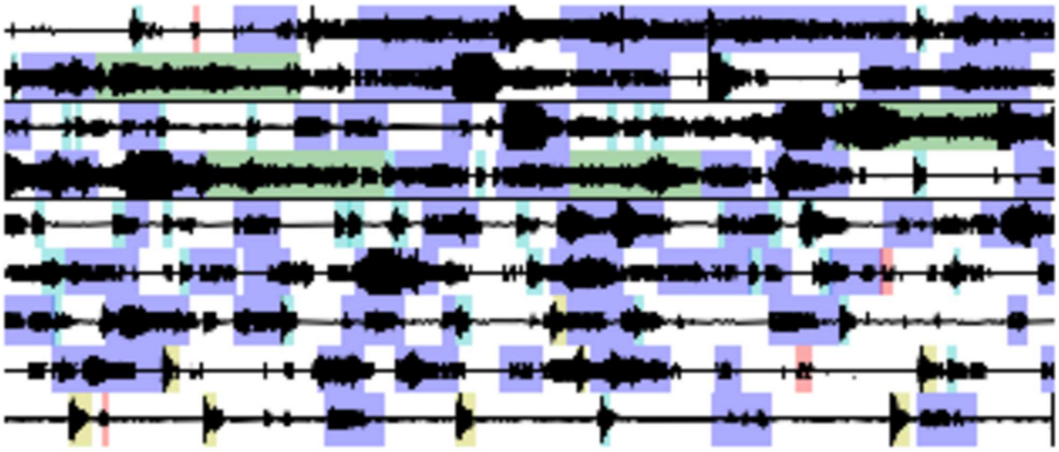

Figure 5.1: Classification results for 10h continuous recordings at Mt. Merapi. Color codes correspond to Hybrids in cyan, PyroclasticDensityFlows in green, Rockfalls in blue, MultiPhase events in red and VolcanoTectonic in yellow.

of different signals and signal amplitudes, HMMs are able to distinguish the target classes from each other. This allows the question whether they could be applied to completely different scenarios (more details on the Mt. Merapi data set see appendix 5.7). Should not the data from a station with high noise level look similar? The noise may resemble the blue and green events and the target waveform the red or yellow (Fig. 5.1).
In this paper we focus on seismic monitoring data from deep geothermal power-plants. Usually the power-plants are located near urban areas where their operating company can sell the power and more important heat and where we experience the corresponding high level of anthropogenic noise. Consequently the signal to noise ratio is often low and the small induced earthquakes, which are important for risk assessment, are not easy to detect. Using a short time average / long time average trigger (STA/LTA), a band-limited pre-filtering, in combination with coincidence sums over at least three stations achieves reasonable detection rates. Nevertheless, for monitoring induced seismicity often just a few stations are available and some of them might be inoperational due to technical problems. As a consequence the number of false alarms will be high, even if coincidence of triggers are tested. For these cases, we designed an earthquake detector based on Hidden Markov Models, whose performance rate is similar to the STA/LTA with at least three stations for coincidence sums.

In this paper we try to give an informal and quite natural introduction to HMMs. In the rest of the paper we concentrate on the system refinement. The procedures, how to adjust the HMM classification to a probabilistic earthquake detector, are explained in a cookbook like style with many demonstrative examples. The proposed probabilistic detector is applied to continuous seismic monitoring data of a geothermal power-plant and the results are compared to those of a recursive STA/LTA with coincidence sums.

5.2 From Gaussian Classifiers to Hidden Markov Models

In this section we provide an informal HMM background for the application and the practically more important issue of system refinement, during which the speech recognition models are actually adapted for seismology. In order to build a detector we simply recast the detection problem into a classification problem with two classes, `noise` for noise and `induced` for induced seismicity.

For the classification, the seismic signal is not represented by the waveform itself but by a set of characteristic functions, which better discriminate the different classes (for more details on features and their selection process, see the studies on discrete Hidden Markov Models Ohrnberger, 2001; Beyreuther and Wassermann, 2008a). These characteristic functions are transformed to their principal axis and called features in the following. The used features are shown in Fig. 5.2. Each event in the training

hob1 (0.47 - 0.78Hz); hob2 (0.70 - 1.2Hz); hob3 (1.1 - 1.8Hz); hob4 (1.6 - 2.6Hz); hob5 (2.4 - 4.0Hz); hob6 (3.6 - 5.9Hz); hob7 (5.3 - 8.9Hz); hob8 (8.0 - 13Hz); hob9 (12 - 20Hz); hob10 (18 - 30Hz); normalized envelope E; instantaneous frequency Z; instantaneous frequency N; centroid time E; centroid time N;

Table 5.1: A list of the used characteristic functions, separated by semicolons. `hob` corresponds to halve octave band, the frequency range is hereby included in brackets. The characteristic functions are transformed to their principal axis, the so called features, which are plotted in Fig. 5.2.

data set corresponds to one line. Due to the transparency of the lines a crossing of multiple lines will result in a more intense color, which allows us to actually see the trend and the variability of the features. The extreme values are visible too, which would be omitted when solely plotting the mean and the variance or the median and the quantiles of the features, respectively.

We focus first on the feature labeled zero (top left in Fig. 5.2). By using all `noise` (gray) feature values (over time), we estimate their probability distribution `b`. In order to build up a classifier, we also need to estimate the probability distribution `b` for the remaining classes (here just the `induced` class in red is missing). In order

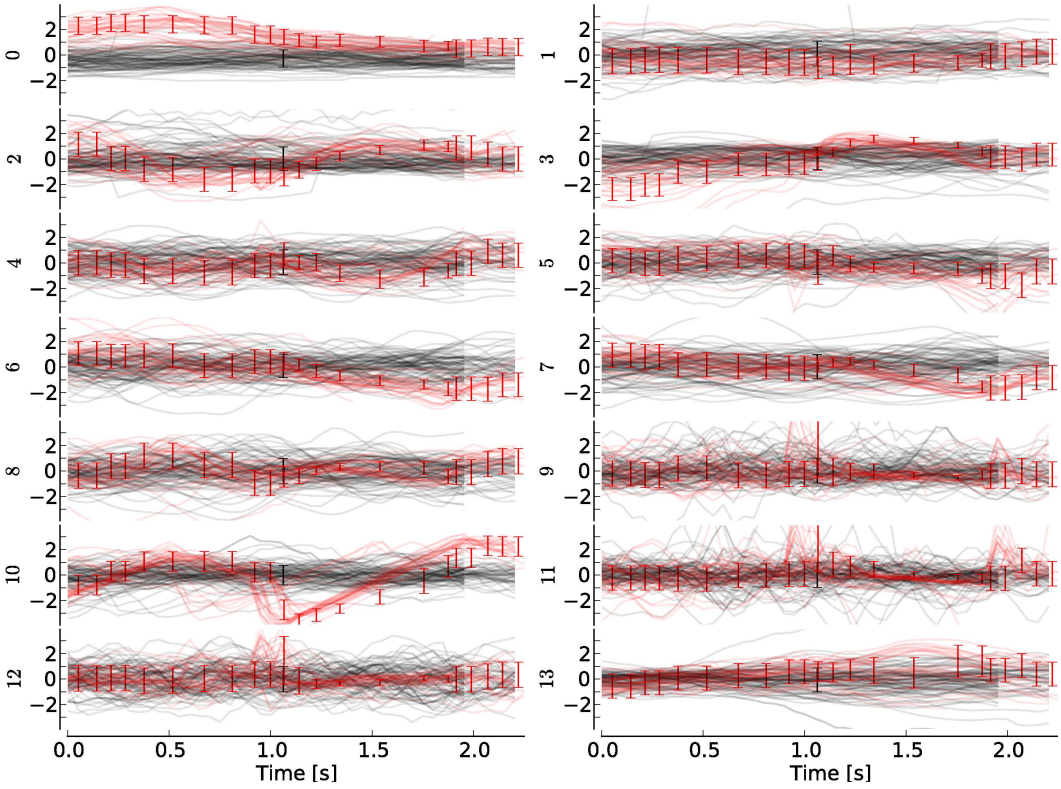

Figure 5.2: The different features used for optimized HMM detection. Each line corresponds to one event in the training data set, the `induced` earthquake events are shown in red, the `noise` events in gray. The features are the principal axis of the characteristic functions given in Tab. 5.1 (the first 14 are shown here). The error-bars correspond to the trained models/Gaussians (see text for details).

to classify an unknown value $\mathtt{o}$, we choose that class, whose distribution $\mathtt{b}$ has the highest probability for the unknown value $\mathtt{o}$ (for visualization see Fig. 5.3). When classifying multiple values over time $\mathtt{t}$, we multiply the probabilities of every individual value $\mathtt{b}(\mathtt{o}_1)\mathtt{b}(\mathtt{o}_2)\mathtt{b}(\mathtt{o}_3)...\mathtt{b}(\mathtt{o}_t)$ and again the most probable class, i.e. the one with the highest product is chosen. However, a single feature might not be enough to distinguish the classes. Therefore the procedure is generalized and applied to all other features, resulting in a 15 dimensional distribution $\mathtt{b}$ and feature vector $\mathbf{o}$. Obviously, the described technique works well for stationary signals. So lets secondly focus on the feature labeled 10 in Fig. 5.2 and examine the `induced` earthquake class (red lines). The events are clearly time dependent, especially the values between 1s and 2s change significantly in time. By estimating only a single distribution $\mathtt{b}$ for all values $\mathtt{o}$ in time, one actually blurs all the time dependent characteristics. Similar problems arise in spectral density estimation; the common approach to avoid this effect is a time-frequency representation (e.g., short term Fourier transform). An comparable idea is used here: Separate distributions $\mathtt{b}_i$ are estimated for individual earthquake segments

i. The different segments are then connected with transition probabilities a_{ij} (from segment i to segment j). The combination of all parameters, i.e. all Gaussians b_i and transition probabilities a_{ij}, is called HMM.

In praxis, the segmentation, i.e. the estimation of the transition probabilities of the earthquake is done automatically by an expectation maximisation algorithm during HMM training. For each class only the segment sequence with the highest probability is taken into account during the actual classification. The segments are called states in the following and the transition probabilities are named state transition probabilities. The distribution b in general might be composed out of multiple Gaussians, which are called Gaussian mixtures.

Fig. 5.2 also shows the estimated probability distributions of b_i after HMM training as error-bars. The center of the error-bar represents the mean, the total hight corresponds to twice the standard deviation (only one mixture is used). The horizontal distance between multiple error-bar corresponds to the mean duration of a state derived from the self transition probabilities a_{ii}. For a more formal and probabilistic description of HMMs, see Rabiner (1989); Young et al. (2002).

5.3 System Refinement

In speech recognition the training corpus/databases is large (160000 telephone utterances in Qing et al., 2000), which allows a robust estimate of even a large number of free HMM parameters. Conversely, in earthquake classification only a low number of training events (Tab. 5.2) is available and statistically an over-training of the HMM parameters is likely. In order to nevertheless achieve robust results, the degrees of

Class	#Events	#DataPoints	median(DataP./event)
loc	21	1009	48.0
noise	90	4180	45.0

Table 5.2: The table shows the number of events for training, their total amount of data-points/time samples and the median amount of data-points per event.

freedom of the HMM parameters (a_{ij}, b_i; $\forall i, j$) is limited. It is crucial to refine the model parameters in such a way, that the HMM best represents the earthquake/event. Therefore a number of system refinement steps is necessary, which are explained in the following.

5.3.1 Left-Right Model Topology

In their general definition HMMs allow transitions from any state to another. For instance in Fig. 5.2, a transition from the last error-bar back to the first error-bar is

possible. However, a "beginning" segment (state) will never follow the "coda" segment (state) of the same earthquake and consequently the model topology is restricted to self transitions and transitions to the next state. The corresponding model topology is called "left-right".

5.3.2 Grand Variance Models

The computation of a robust variance is the most difficult part in the model estimation. Especially a low amount of training samples easily leads to an over-trained (too small variance) and thus useless model. The reason is that the training data set often does not represent all possible variations of the events in reality and so only certain variations are represented by the model.
Figure. 5.3a illustrates the problem. In this figure the dominant frequency content of the events is used as an example. The light gray points correspond to `noise` events, the dark gray points to `induced` events, respectively. For both classes, a Gaussian is estimated with mean 5Hz for the `noise` and 3Hz for the `induced` class. The most probable distribution for the unknown point at 1.5Hz (denoted as a cross) is the distribution `noise` with the mean at 5Hz, even though the point is much closer to the mean/center of the `induced` distribution at 3Hz. This is good if the variance of the `induced` distribution is correct which, however, requires a very high amount of training data. But as soon as the `induced` distribution is over-trained, which means that the estimated variance is smaller than it actually is, the point (cross) is misclassified as `noise`. This effect results from the normalization of the Gaussians to the area of one. Therefore a smaller variance will yield a higher probability around the mean but a lower probability at the sides of the distribution (Fig. 5.3a).
One can avoid that behaviour through grand variance models. By tying the model variances together (Fig. 5.3b) the dark points corresponding to the `induced` class are certainly not described as well by the dark distribution in Fig. 5.3a. Yet, the unknown point (cross) is classified as the `induced` class because it has the closest Gaussian center (for details on grand variance models, see Young et al., 2002, p. 152).

5.3.3 Number of States

How to choose the optimal number of states for a HMM is always under debate. A higher number of states allows a much better time discretization in the feature space, thus allowing a better description of the event. But this also has the effect that the system is easily over-trained (due to the high degree of freedom) and achieves poor results on unknown data. Our twist, how to allow a high number of states and at the same time keep the degree of freedom low, is introduced next.

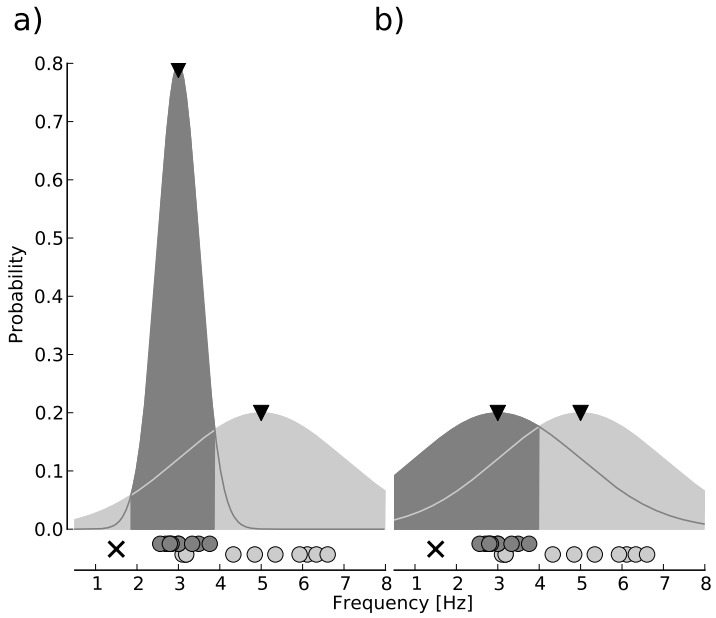

Figure 5.3: a) Two Gaussians estimated from the dark and light gray points, the centers of the Gaussians are marked as black triangles. Even though the unknown point (cross) is closer to the center of the dark gray distribution, the most probable distribution is the light gray one. This can be circumvented as shown on the right. b) The two Gaussians are estimated with tied variances. The best distribution for the unknown point (cross) changes to the closest distribution center, which is now the dark gray one.

5.3.4 Clustering

In order to further restrict the number of free parameters without loosing too much time discretization, similar Gaussians (with respect to their euclidean distance) are tied together with a clustering approach. In the training tied parameters behave like a single one and so effectively reduce the number of free parameters. So even if there is a high number of states, only the number of states after clustering need to be trained. The key parameter for clustering is the target number of states one wishes to have after clustering, which can be estimated in one of the following ways.

The most intuitive possibility is to count the minimum number of segments one would use to describe the earthquake in the feature space (Fig. 5.2) and use this as the target number of Gaussians. The second possibility is to plot the cluster threshold distance versus the number of states (Fig. 5.4a) and to select the number of clusters with the scree test. A low value of the cluster threshold distance means that only those Gaussians are tied together, whose distance between their centers is less or equal the given low value of the cluster distance. Consequently a low cluster threshold will result in a high number of Gaussians. In our case we want to tie similar Gaussians while keeping the prominent ones. We may select 15 cluster Ids.

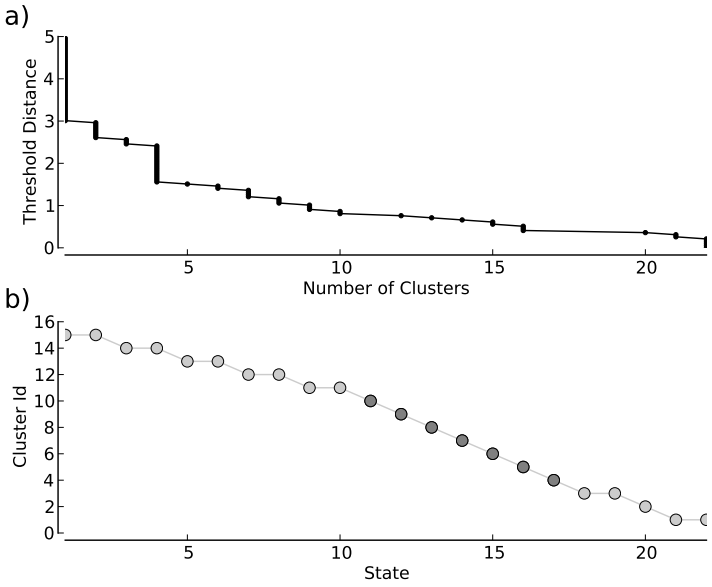

Figure 5.4: a) Cluster threshold distance versus the number of states. b) Cluster Id versus total number of states: circles/states with the same Cluster Id are tied together. Light gray points correspond to states which have a grand variance, for the dark gray points the variances are not tied.

One important side-effect of tied states is that it provides the possibility to introduce a minimum duration for a certain state. Fig. 5.4b shows for instance that the first state has a minimum length of two time samples. Of course also longer durations are allowed through the possible transition to the same earthquake segment (state). However, the actual problem becomes obvious in this example: Consider a HMM which segments the earthquake in three states and a 20 time-samples long feature vector to classify. What we actually see in praxis is that in the classification this state sequence is taken: $(1 \rightarrow 2 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow ... \rightarrow 3)$. So the 20 time-sample long signal is mostly represented by the third state (the state transition probabilities a_{ij} are of minor importance in the presence of the probabilities of the multidimensional distribution b_i). In order to avoid the collapse to the third model state, a minimum duration can be included which inserts new states and ties them to the original one. So for a minimum duration of 5 time-samples per state, the number of states needs to be increased to $5 \cdot 3 = 15$ and 5 consecutive states needs to be tied to together. The clustering and the resulting tied states (Fig. 5.4) provided a data driven way to set the minimum length of each state respectively.

5.3.5 Partly untying the Grand Variance

A better discrimination than through the grand variance models can be achieved by untying the variances for those time sections of the earthquake, which are not prone to

over-training. In the feature plot Fig. 5.2 the variance is quite high at the beginning and the end of the earthquake and low around 1-2s. This observation is exploited by untying the variances of the states corresponding to the 1-2s time section. In Fig. 5.4b, the states with a global variance are indicated in light gray, the points in dark gray correspond to states whose variances are not tied.

5.3.6 Cutting the Training Data to same Length

In speech recognition the end of a certain word is quite clearly defined. In seismology the variability of the event-length for the same earthquake class is high and strongly depends on the coda. The most obvious strategy is to select the length of the event based on the signal to noise ratio. However, this strategy is problematic as we demonstrate with the following example (Fig. 5.5): The median length of the `induced` class

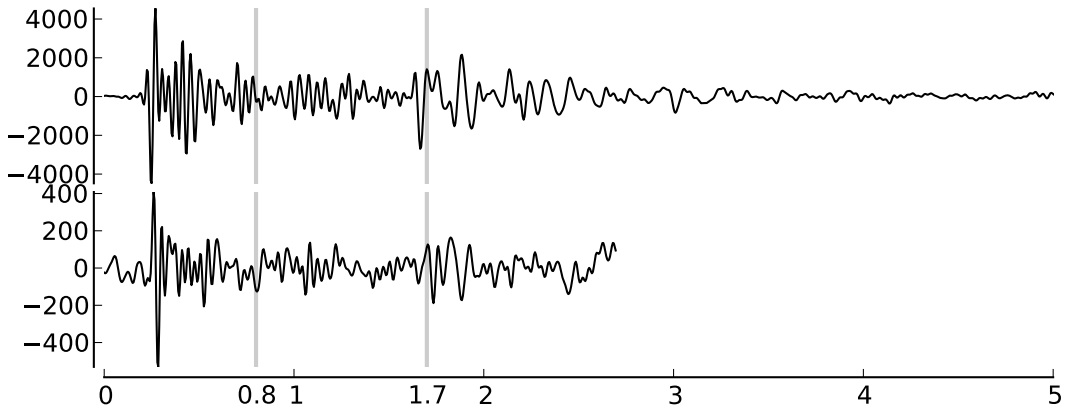

Figure 5.5: Two events whose length is determined such that all values are above signal to noise ratio two. Note that the upper event is twice as long as the lower one. The vertical gray lines indicate a segmentation of the events in three parts.

is about 2.5s, but there are high SNR samples in the training data set of up to 5s — which contains more coda. Assume a 3 state model with equipartition in time, with the last state corresponding to the coda. For the 2.5s event (lower plot) state 1, 2, 3 corresponds to approximately 0-0.8s, 0.8-1.7s, 1.7-2.5s respectively. For the 5s earthquake there are only 1.7s training data for the first two states. For the third state there are 3.3s, the same amount as 2 earthquakes would deliver for the first two states. The expectation maximisation algorithm will over pronounce the coda in the model. Consequently restricting the events to a maximum length is a crucial and unintuitive point in the parameter estimation.

5.3.7 Noise

The time dependency is relevant only for the causal/time dependent earthquake events. This time dependence is not needed in the case of the noise model, since noise most commonly does not have a causal or time dependent structure which could be described by different states. That is why we use a single state HMM (also called General Mixture Model) for the noise. The exact parametrization of the noise and the induced class is given in the following section where the probabilistic earthquake detector is applied.

5.4 Application

In this section, the probabilistic earthquake detector is applied to anthropogenic induced seismicity. We think that this application is very promising for evaluating the performance for the following reasons:

1.) For monitoring geothermal power-plants usually just few stations are used. Typically there are time intervals where some stations are inactive due to technical difficulties, resulting in just a few stations that are available for detection and classification. For the detection normally a STA/LTA in combination with coincidence sums might be the method of choice. However, in cases where there are e.g. only two stations available, the coincidence sums will not be able to remove noise triggers completely. The probabilistic earthquake detector, on the other hand, operates on a single station and allows detection rates similar to classic methods which combine multiple stations.

2.) For an earthquake detection and classification system, the false alarm rate is a crucial factor. In one month there are approximately half a million noise classification versus just a few earthquakes. So a realistic noise environment with all sorts of noise sources is needed for a robust performance measure, especially noise events with a similar duration as earthquakes. This is given in particular near geothermal power-plants, which are usually located close to urban areas with the typical low signal to noise ratio and all kinds of different noise sources (e.g. trucks, pumps, trees).

The two classes, induced earthquakes `induced` and noise `noise` served as examples in the theory section where their features were shown (Fig. 5.2). The trained HMMs for both classes are shown as error-bars in the same figure. In the following we give a short description of the parameter constellation, which we used.
Tab. 5.2 showed the number of training events and the corresponding amount of training samples. In order to better describe the time dependency, an initial induced earthquake model with 22 states is used (approximately half of the median number of data points/time samples per training event see Tab. 5.2) which are then clustered and tied to 15 independent Gaussians. As a consequence 15 means and variances need to be estimated from 1009 samples, which result in just about 32 samples per parameter, or 1.5 parameters per event in the training data set. For robust classification we rely on the grand variance models, where the variance of the earthquake data is high (light

	# of Events	Median. Conf.	Conf.>1000
missed earthquakes	3		4
correct earthquakes	18	32046.3	17
	18/21 ≈ 86%		17/21 ≈ 81%
correct noise	2270364		2270390
false alarms	32	15.0	8
averaged over 3.9 month	0.3 per day		0.07 per day

Table 5.3: Classification results for a four month period starting on 1. Januar 2010

gray points in Fig. 5.4, see also Fig. 5.2) and we only untie the variance in the middle part of 1-2s where the waveforms are really sharp and well defined (dark gray points Fig. 5.4, see also Fig. 5.2).
For the noise model, we use a single state HMM with only one Gaussian mixture. We tried a maximum of 10 Gaussian mixtures but overall the results did not change dramatically.

For the performance measure, a 3.9 month period was selected. During this period two other stations were also active which allowed us to verify the classification results by locating the events or, if that was impossible, by inspecting the waveforms of different stations.
For the classification, a 9s sliding window is used with a step of 4.5s. In total 2270400 classifications are made, the results of which are shown in Tab. 5.3. For each window a [`noise`, `induced`, `noise`] and a only [`noise`] sequence is classified, which allows to calculate the confidence level $\mathtt{P}(\mathtt{induced})/\mathtt{P}(\mathtt{noise})$ (with $\mathtt{P}$ being the probability). Similar to association rules (Hastie et al., 2009, p. 492ff) or the trigger thresholds we set the minimum confidence level and the minimum log probability of a detection to 3 and -8 respectively. Tab. 5.3 clearly suggests that a further restriction of the confidence interval will result in a better classification result. By considering only classifications whose confidence level is above 1000, the false alarm rate per day could be reduced significantly from 0.3 to 0.07 false alarms per day.

In order to get a reference for these performance rates, a recursive STA/LTA in combination with coincidence sums was applied to the same time period. Two other stations were deployed, one was not available for the first 26 days. The STA/LTA window length was set to 0.5s and to 10s for the STA and LTA window respectively and custom thresholds for the trigger on and off values were set for each station individually. In order to achieve reasonable detection rates (Tab. 5.4) it was necessary to pre-filter the waveforms to 10-20Hz and build a coincidence of all available station (2 for the first 26 days and 3 for the remainder). Especially when there are only two stations available, the false alarm rate increases dramatically, and all the false alarms need to be laborious verified by interactive analysis.

	2 stations	3 stations
missed earthquakes	1	0
correct earthquakes	9	11
false alarms	434	28
false alarms per day	$434/26 = 16.7$	$28/(3 \cdot 30) = 0.31$

Table 5.4: Results of a bandlimited STA/LTA for same time span as Tab. 5.3. During the first 26 days, only 2 stations were active, that is for building coincidence sums 2 respectively 3 stations were used.

5.5 Conclusions

In this study we show the design of a probabilistic earthquake detector which operates on a single station. Despite of this limitation, the applied method has only slightly lower performance than a pre-filtered STA/LTA trigger with coincidence sums of 3 stations. We applied the detector to induced earthquakes from geothermal power-plants for a 3.9 months period. One of the reference stations was inactive for the first 26 days and immediately the false alarm rate of the STA/LTA trigger with only two stations for the coincidence sums rises dramatically. The STA/LTA performs already bad with two stations available, however, with only one station it is even impossible to extract the events because of nearly continuous triggering. For these cases the probabilistic earthquake detector provides a promising and valuable alternative.

The architecture of the probabilistic earthquake detector is based on HMM, a technology borrowed from speech recognition. Even though earthquake signals look similar to speech signals, several pitfalls need to be circumvented in order to apply the HMM in this field. The main difference is the small waveform training databases in seismology and the strongly varying durations of certain earthquake segments (e.g. coda). Throughout the paper we concentrated on explaining the adaption steps and showed many examples, such that the taken steps could be used as basis for building other earthquake detectors or classifiers (appendix 5.7 shows details of the application of the above described system to seismic data of Mt. Merapi).

5.6 Acknowledgements

The authors wish to thank Heiner Igel for his kind support. We also wish to thank the authors of the HMM Tool Kit (HTK) Young et al. (2002) for providing and developing such a powerful software package. This work was partially funded by the German Ministry for Education and Research (BMBF), GEOTECHNOLOGIEN grant 03G0646H.

5.7 Appendix: Application to Seismic Data of Mt. Merapi

For the sake of completeness we want to give some more details of the classification system shown in Fig. 5.1. There are six different classes Hybrids (HY), PyroclasticDensityFlows (PDF), Rockfall (GU), MultiPhase (MP) and VolcanoTectonic (VT) events. The size of the training data set is given in Tab. 5.5, the characteristic functions that were used are given in Tab. 5.6 (for more details on the features and their selection in case of Mt. Merapi, see Ohrnberger, 2001). Similar to the system refinement section and the main application, we construct a classifier. We use a class dependent window length: `HY` 51s, `PDF` 383s, `GU` 146s, `MP` 52s, `UN` 37s, `VT` 90s. The window step is 90s. The HMM parameter constellation is given in Tab. 5.7 and in the following. Four

Class Name	HY	PDF	GU	MP	VT	noise
# Events	32	19	73	30	30	65
# DataPoints	4364	15533	35141	3478	4988	31420
median(DataPoints/event)	137	872	479	117	167	318

Table 5.5: The table shows the number of events for training, their total amount of data-points/time samples and the median amount of data-points per event.

hob1 (0.35-0.59Hz); hob2 (0.52-0.88Hz); hob3 (0.80-1.3Hz); hob4 (1.2-2.0Hz); hob5 (1.8-3.0Hz); hob6 (2.7-4.4Hz); hob7 (4.0-6.7Hz); hob8 (6.0-10Hz); normalized envelope Z; instantaneous bandwidth Z; centroid time Z;

Table 5.6: A list of the used characteristic functions, separated by semicolons. `hob` corresponds to halve octave band, the frequency range is hereby included in brackets. The characteristic functions are transformed to their principal axis, which are then called features and used as input for the HMM training.

Class Name	HY	PDF	GU	MP	VT
# States	52	325	190	44	64
# Clusters	13	8	10	17	12

Table 5.7: Cluster design for the Mt. Merapi data set. The total number of states per class are given in the first row. The second row the numbers of clusters are given, which correspond to the number of independent states after clustering.

Gaussian mixtures were used for the `noise` distribution `b`, and the variances for `PDF`, `GU` and `noise` were tied together (grand variance).

The system was evaluated on 3.6 months of continuous seismic data. The outcome is presented in Fig. 5.6 which shows the results of the HMM classification versus the interactive event list of the local volcanologists. Note that especially the GU events

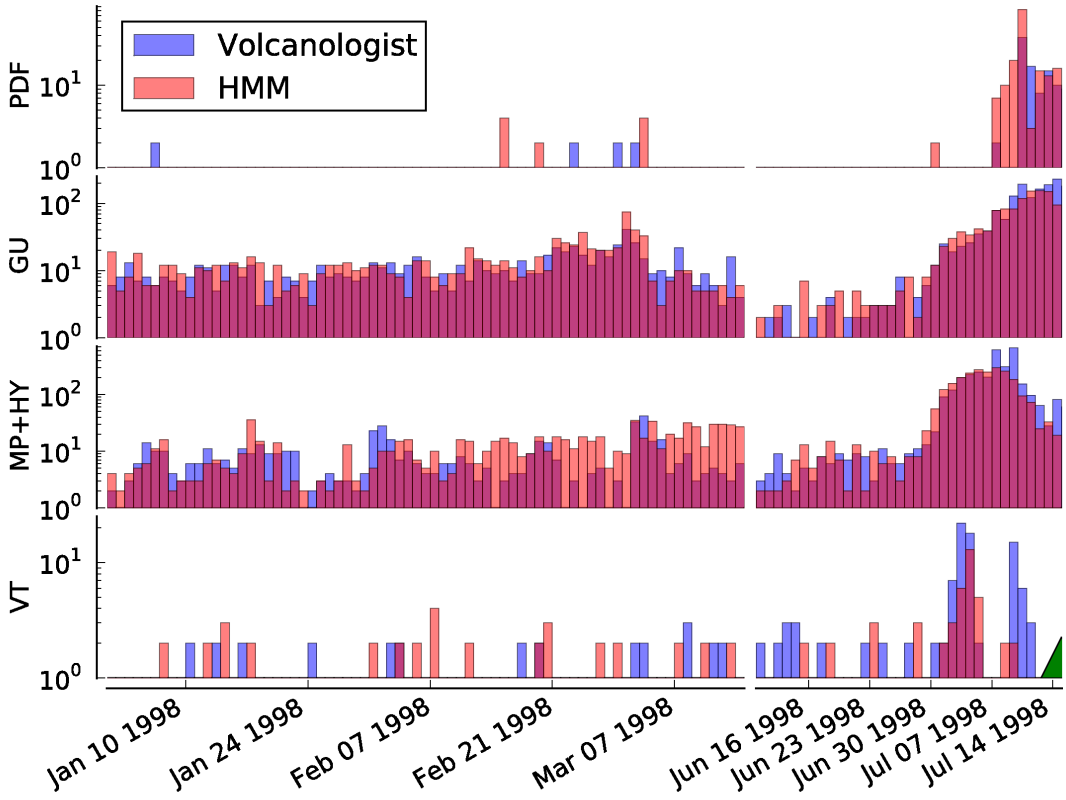

Figure 5.6: Continuous classification results for seismic events from Mt. Merapi. The light red bars correspond to the classification results of the HMM classification, the blue bars correspond to the classifications of the local volcanologists. The period starts on 1998-01-01, the station was inoperable between 1998-03-14 and 1998-06-10. The green triangle marks the onset of an eruption. The bars are transparent, thus a blue bar overlaid with a light red bar results in a purple bar.

match well the event rates classified by the local volcanologists. The MP events perform reasonable in general, and near the onset of the eruption especially well (green triangle in Fig. 5.6). However, there are still major differences in the VT and PDF rates, which are partly due to the model design on the one hand and might be due to slight inconsistencies in the classification behaviour of the local volcanologists in the other hand. Note also, that the HMMs are operating on a single channel, whereas the volcanologists compare the waveforms of several stations in order to set a certain class type. All training data are from time periods after July first 1998. Another promising observation is that even though that the waveforms might change slightly in time the classifications results from January till mid of March still match the event rates of the volcanologists well.

6 Conclusion and Outlook

The main goal of this thesis was to develop a prototype automatic detection and classification system for earthquakes based on up-to-date speech recognition algorithms which are known as Hidden Markov Models (HMMs). Besides the necessary adaptations concerning pure data-formats and characteristic functions, the research of this thesis mostly focused on the HMM design in regard to the inherent characteristics of earthquakes. Especially the variation of earthquake durations, the limited amount of training data and the extremely long periods of noise play an important role. The development of the prototype system was devided in four major parts. As a first step a discrete HMM (DHMM) prototype system was developed, followed by a more general HMM implementation. A flexible duration was realized in a compound prototype consisting of Hidden semi-Markov Models (HSMMs) and Weighted Finite-State Transducers (WFSTs) whose design was finally approximated through a much faster performing HMM approach. The results of these four prototypes are discussed in the following.

The DHMM prototype (Beyreuther and Wassermann, 2008a) was applied to classify three different types of earthquakes in one month of continuous seismic data (10368000 classifications, 0.25s window step size) at Mt. Hochstaufen (Germany). The results of three stations were combined and compared to a standard recursive STA/LTA detection system approach. When comparing only the number of detected events, which underestimates the performance of the DHMM method as it not only detects but also classifies, the recursive STA/LTA detection performs slightly better with a 90% detection rate compared to the DHMM detection and classification with 82% respectively. However, when comparing the DHMM to other classification systems (e.g. support vector machines or artificial neural networks), where the erroneous detection rate of the pre-detector needs to be included, the DHMM detection and classification certainly shows comparable results.

The more general HMM prototype (Beyreuther et al., 2008b) was applied in order to automatically detect as well as discriminate earthquakes of volcanic and non-volcanic origin at the Teide-Pico Viejo volcanic complex on Tenerife (Azores). For evaluation purpose one month of continuous seismic data was processed (324000 classifications, 8s window step size). The HMM classification results of a single seismic station nearly match the number of earthquakes published in the IGN earthquake catalogue. When using only earthquakes which were marked by the proposed HMM as high confident earthquakes, 86% of the earthquakes of the catalogue with a signal to noise ratio greater than two could be detected. Especially the low false alarm rate with 0.1 false alarms

per day is remarkable which would allow e.g. to automatically deduce earthquake statistics or early warning.

The combined HSMM and WFST prototype (Beyreuther and Wassermann, 2011a) was applied to another month of continuous seismic data (370285 classifications, 7s window step size) at Mt. Hochstaufen, where the length of the earthquake classes of interest varies from 2s - 120s. In order to emphasize the performance gain of the HSMM, the same period was reclassified with the previously proposed HMM. The application of the more general model and classifier design of the HSMM/WFST prototype allowed a 40% better classification rate than the previously introduced HMM method. However, the CPU time also increased significantly to finally 0.5h of CPU time for 1.0h of data.

The HMM approximation of the combined HSMM and WFST prototype (Beyreuther et al., 2011b) was applied to detect earthquakes induced by geothermal power-plants in 3.9 month of continuous seismic data (2270400 classifications, 4.5s window step size). When using a single station, the applied method shows only slightly lower performance in comparision to a pre-filtered STA/LTA trigger with coincidence sums of 3 stations. For the latter, however, one of the reference stations was inactive for the first 26 days and immediately the false alarm rate of the STA/LTA trigger with only two stations active rose above the false alarm rate of the HMM.
As a proof of concept the same HMM parameter design was applied to classify 3.6 month continuous volcano seimic data (103680 classifications, 90s window step size) from Mt. Merapi (Indonesia). In total six different types of volcano seismic events were classified and in order to evaluate the results, the event rates were compared to the interactive classifications made by local volcanologists. While some of the earthquake types perform especially well, others show still considerable differences. These differences are most likely caused by the model design and/or they might be due to slight inconsistencies in the classification behaviour of the local volcanologists.

In summary, the main objective of this thesis was the improvement of the adaption of HMM design and algorithms to earthquake data. The different HMM prototypes were successfully applied to various continuous seismic data sets. In total 10.5 month of continuous seismic data was automatically detected, classified and evaluated. This corresponds to approximately 13 million classifications. The overall performance clearly indicates that the automatic HMM based detection and classification is applicable in seismology as it shows at least similar results as alternative methods in various applications. Especially when there are only a few stations available, when the signal to noise ratio is low or when the classification of many different event types is needed, the automatic HMM based detection and classification outperforms the mentioned alternative methods.

Future work needs to focus on the combination of different stations, which should improve the results in a comparable manner as coincidence sums or binder modules do for trigger algorithms. The differences in the onset times, the probabilities and confidence measures for each earthquake class and the event length could be used as

input for support vector machines or tree based classifiers, which then decide whether an event occurred or not. However, even simple coincidence sums could already improve the results significantly.
Also applications at other volcanoes or seismic active areas are important in order to establish the HMM based detection and classification as state of the art automatic earthquake detection and classification system. Especially the new insights and the distinctiveness of other datasets will help to further develop the algorithms and the most favorable HMM design.

The design of the extensive framework code is general and parts of it have been constantly released to the ObsPy toolbox (Beyreuther et al., 2010). Nevertheless, much expert and theoretical knowledge is still necessary in order to apply the HMM models to a new area / volcano / dataset. Determining the best HMM design automatically will allow non speech recognition experts to apply the system, too. Especially a generally applicable set of characteristic functions would be helpful.
Also the model design still holds room for improvements. Discriminate training (e.g. the `HMMIRest` program of HTK) of the HMMs will probably improve the results even further and the introduction of pruning thresholds in the classification algorithm will allow even faster processing times. The HSMM and WFST approach can be further developed, too. The flexibility of the WFSTs allows to easily constrain, to customize and to adapt the decoding process to the specialities of seismology. Using faster WFST decoders like e.g. `juicer` (`http://juicer.amiproject.org/juicer`) or `openfst` (`http://www.openfst.org`) could make this approach much more attractive.

In order to compare the results of this thesis to other methods in earthquake detection and classification (e.g. artificial neuronal networks, support vector machines, self organizing maps or tree based classifiers) benchmark data sets are required. Unfortunately these are currently not available for the field of seismology and, concluding this thesis, there is a strong demand for them.

Bibliography

Alasonati, P., Wassermann, J., Ohrnberger, M., 2006. Signal Classification by Wavelet-Based Hidden Markov Models: Application to Seismic Signals of Volcanic Origin. In: Mader, H., Connor, C., Coles, S. (Eds.), Statistics in Volcanology. Geological Society of London, pp. 161—174.

Allen, R., 1982. Automatic phase pickers: Their present use and future prospects. Bulletin of the Seismological Society of America 72 (6B), 225—242.

Baer, M., Kradolfer, U., 1987. An automatic phase picker for local and teleseismic events. Bulletin of the Seismological Society of America 77, 1437—1445.

Barnes, A. E., 1993. Instantaneous spectral bandwidth an dominant frequency with applications to seismic reflection data. Geophysics 58, 419—428.

Barsch, R., 2009. Web-based technology for storage and processing of multi-component data in seismology. Ph.D. thesis, LMU München.

Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y., Wassermann, J., 2010. ObsPy: A Python Toolbox for Seismology. SRL 81 (3), 530—533.

Beyreuther, M., Carniel, R., Wassermann, J., 2008b. Continuous Hidden Markov Models: Application to automatic earthquake detection and classification at Las Cañadas caldera, Tenerife. Journal of Volcanology and Geothermal Research 176 (4), 513—518.

Beyreuther, M., Hammer, C., Wassermann, J., Ohrnberger, M., Megies, T., 2011b. Design of a Probabilistic Earthquake Detector. GJI, Submitted to Geophysical Journal International, 09/2010.

Beyreuther, M., Wassermann, J., 2008a. Continuous earthquake detection and classification using Discrete Hidden Markov Models. Geophysical Journal International 175 (3), 1055–1066.

Beyreuther, M., Wassermann, J., 2011a. Hidden semi-Markov Model based earthquake classification system using Weighted Finite-State Transducer. Nonlin. Processes Geophys. 18, 81—89.

Carniel, R., Cecca, M. D., Rouland, D., 2003. Ambrym, Vanuatu (July-August 2000): Spectral and dynamical transitions on the hours-to-days timescale. Journal of Volcanology and Geothermal Research 128, 1—13.

Carniel, R., Tárraga, M., 2006. Can tectonic events change volcanic tremor at Stromboli? Geophysical Research Letters 33, L20321.

Carniel, R., Tárraga, M., Barazza, F., García, A., 2008a. Possible interaction between tectonic events and seismic noise at Las Cañadas volcanic caldera, Tenerife, Spain. Bulletin of Volcanology 70 (9), 1113—1121.

Carniel, R., Tárraga, M., Jaquet, O., Ortiz, R., García, A., 2008b. The seismic noise at Las Cañadas volcanic caldera, Tenerife, Spain: Persistence characterization, and possible relationship with regional tectonic events. Journal of Volcanology and Geothermal Research 173, 157—164.

Creager, K., 1997. CORAL. Seism. Res. Lett. 68 (2), 269—271.

Del Pin, E., Carniel, R., Tárraga, M., 2008. Event recognition by detrended fluctuation analysis: An application to Teide-Pico Viejo volcanic complex, Tenerife, Spain. Chaos, Solitons and Fractals 36 (5), 1173—1180.

Fattori Speranza, F., Carniel, R., 2008. Structural changes of volcanic tremor at Stromboli volcano. Journal of Volcanology and Geothermal Research 171, 103–117.

Gentili, S., Bragato, P., 2006. A neural-tree-based system for automatic location of earthquakes in northeastern italy. Journal of Seismology 10, 73—89.

Gottsmann, J., Carniel, R., Coppo, N., Wooller, L., Hautmann, S., Rymer, H., 2007. Oscillations in hydrothermal systems as a source of periodic unrest at caldera volcanoes: Multiparameter insights from Nisyros, Greece. GRL 34, 7307–+.

Gudmundsson, O., Finlayson, D. M., Itikarai, I., Nishimura, Y., Johnson, W. R., 2004. Seismic attenuation at Rabaul volcano, Papua New Guinea. Journal of Volcanology and Geothermal Research 130, 77—92.

Hanka, W., Kind, R., 1994. The GEOFON Program. Annali di Geofisica 37 (5), 1060—1065.

Harris, M., Young, C., 1997. MatSeis: A SEISMIC GUI AND TOOLBOX FOR MATLAB. Seism. Res. Lett. 68 (2), 267—269.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical learning: data mining, inference and prediction, 2nd Edition. Springer.

Havskov, J., Ottemöller, L., 1999. SEISAN EARTHQUAKE ANALYSIS SOFTWARE. Seism. Res. Lett. 70 (5), 522—528.

Hoffmann, W., Kebeasy, R., Firbas, P., Jun. 1999. Introduction to the verification regime of the Comprehensive Nuclear-Test-Ban Treaty. Physics of the Earth and Planetary Interiors 113, 5–9.

Hunter, J. D., 2007. Matplotlib: A 2d graphics environment. Computing in Science and Engineering 9 (3), 90—95.

Ibáñez, J. M., Benítez, C., Gutiérrez, L. A., Cortés, G., García-Yeguas, A., Alguacil, G., 2009. The classification of seismo-volcanic signals using hidden markov models as applied to the stromboli and etna volcanoes. Journal of Volcanology and Geothermal Research 187 (3-4), 218 — 226.

Ibs-von Seht, M., 2008. Detection and identification of seismic signals recorded at krakatau volcano (indonesia) using artificial neural networks. Journal of Volcanology and Geothermal Research 176 (4), 448—456.

Johnson, C. E., Bittenbinder, A., Bogaert, B., Dietz, L., Kohler, W., 1995. Earthworm : A flexible approach to seismic network processing. Iris Newsletter 14 (2), 1—4.

Jolly, G., Scott, B., 2007. Quantifying the risk to scientists monitoring active volcanoes. XXIV IUGG Assembly, Session VS013 Quantifying and expressing volcanic risk: a challenge for the Millennium, Perugia, Italy 2- 13 July 2007.

Joswig, M., 1990. Pattern recognition for earthquake detection. Bulletin of the Seismological Society of America 80 (1), 170—186.

Joswig, M., 1994. Knowledge-based seismogram processing by mental images. IEEE Transaction on Systems, Man, and Cypernetics 24 (3), 429—439.

Joswig, M., 1996. Pattern recognition techniques in seismic signal processing. In: M. Garcia-Fernandez, G. Z. (Ed.), 2. workshop on Application of Artificial Intelligence Techniques in Seismology and Engineering Seismology. Vol. 12. pp. 37—56, cahiers du Centre Europeen de Geodynamique et de Seismologie.

Juang, B.-H., Rabiner, L. R., 1985. A probabilistic distance measure for hidden markov models. AT&T System Technical Journal 64, 391—408.

Kanasewich, E. R., 1981. Time Sequence Analysis in Geophysics, 3rd Edition. The University of Alberta Press, Edmonton, Alberta, Canada.

Kehagias, A., Fortin, V., 2006. Time series segmentation with shifting means hidden markov models. Nonlinear Processes in Geophysics 13 (3), 339–352.

Kittler, J., Young, P., 1973. A new approach of feature selection based on the karhunen-loeve expansion. Pattern Recognition 5, 335—352.

Knill, K. M., Young, S. J., 1994. Speaker dependent keyword spotting for accessing stored speech. Tech. rep., CUED/F-INFENG/TR 193, Cambridge University Engineering Dept.

Langer, H., Falsaperla, S., 2003. Seismic monitoring at stromboli volcano (italy): a case study for data reduction and parameter extraction. Journal of Volcanology and Geothermal Research 128, 233—245.

Langer, H., Falsaperla, S., Masotti, M., Campanini, R., Spampinato, S., Messina, A., 2009. Synopsis of supervised and unsupervised pattern classification techniques applied to volcanic tremor data at Mt Etna, Italy. Geophysical Journal International 178, 1132–1144.

Langer, H., Falsaperla, S., Powell, T., Thompson, G., 2006. Automatic classification and a-posteriori analysis of seismic event identification at soufrière hills volcano, montserrat. Journal of Volcanology and Geothermal Research 153, 1—10.

Leonard, M., 2000. Comparison of manual and autoatic onset time picking. Bulletin of the Sesmological Society of America, 90 6, 1384—1390.

Malone, S., 1997. Electronic Seismologist goes to FISSURES. Seism. Res. Lett. 68 68 (4), 489–492.

Manga, M., Brodsky, E., 2006. Seismic triggering of eruptions in the far field: Volcanoes and geysers. Annual Review of Earth and Planetary Sciences 34 (1), 263—291.

Martí, J., Gudmundsson, A., 2000. The Las Cañadas caldera (Tenerife, Canary Island): an overlapping collapse caldera generated by magma-chamber migration. Journal of Volcanology and Geothermal Research 103, 161—173.

Martí, J., Mitjavila, J., Araña, V., 1994. Stratigraphy, structure and geochronology of the Las Cañadas caldera, (Tenerife, Canary Islands). Geological Magazine 131, 715—727.

McNutt, S. R., 2000. Volcanic seismicity. In: Encyclopedia of Volcanoes. Academic Press, San Diego, pp. 1015—1033.

Milner, K., Thorsten, W., March 2009. New Software Framework to Share Research Tools. EOS 90 (12), 104, http://geosys.usc.edu/projects/seatree.

Mohri, M., Pereira, F., Riley, M., 2002. Weighted finite-state transducers in speech recognition. Computer Speech & Language 16 (1), 69—88.

Mohri, M., Pereira, F. C. N., Riley, M., 1997. General-purpose Finite-State Machine Software Tools. http://www2.research.att.com/ fsmtools/fsm/index.html.

Myers, C., Rabiner, L. R., Rosenberg, A. E., Dec 1980. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. IEEE Trans. on Acoustics, Speech, and Signal Processing ASSP-28 (6), 623—635.

Ohrnberger, M., 2001. Continous automatic classification of seismic signals of volcanic origin at Mt. Merapi, Java, Indonesia. Ph.D. thesis, Institut für Geowissenschaften, Universität Postdam.

Olsen, K. B., Ely, G., 2009. WebSims: A Web-based System for Storage, Visualization, and Dissemination of Earthquake Ground Motion Simulations. Seismol. Res. Lett 80, 1002—1007.

Ortiz, R., García, A., Astiz, M., 2001. Instrumentación en volcanología. Servicio de Publicaciones del Cabildo Insular de Lanzrote, Madrid, 345–347.

Oura, K., Zen, H., Nankaku, Y., Lee, A., Tokuda, K., 2006. Hidden semi-Markov model based speech recognition system using weighted finite-state transducer. In: Acoustics, Speech and Signal Processing. Vol. 1. pp. 33–36.

Oura, K., Zen, H., Nankaku, Y., Lee, A., Tokuda, K., 2008. A Fully Consistent Hidden Semi-Markov Model-Based Speech Recognition System. IEICE Trans Inf Syst E91-D (11), 2693–2700.

Qing, G., Yonghong, Y., Zhiwei, L., Baosheng, Y., Qingwei, Z., Jian, L., 2000. Keyword spotting in auto-attendant system. In: Sixth International Conference on Spoken Language Processing. pp. 1050—1052.

Rabiner, L. R., 1989. A tutorial on Hidden Markov Models and selected applications in speech recognition. Proceedings of the IEEE 77 (2), 257—286.

Scherbaum, F., Johnson, J., 1993. PITSA, Programmable Interactive Toolbox for Seismological Analysis. Tech. rep., IASPEI Software Library, Vol. 5.

Schukat-Talamazzini, E. G., 1995. Automatische Spracherkennung. Vieweg Verlag, Braunschweig.

Sleeman, R., van Eck, T., 1999. Robust automatic p-phase picking: an on-line implementation in the analysis of broadband seismogram recordings. Physics of the Earth and Planetary Interiors 113 (1), 265—275.

Stammler, K., 1993. SeismicHandler - Programmable multichannel data handler for interactice and automatic processing of seismological analyses. Comp. & Geosciences 19 (2), 135—140.

Statistisches-Bundesamt, 2006. Verkehr; verkehrsunfälle. Tech. Rep. 2080700057004, Statistisches Bundesamt, Wiesbaden, fachserie 8 Reihe 7.

Taner, M. T., Koehler, F., Sheriff, R. E., 1979. Complex seismic trace analysis. Geophysics 44 (6), 1041—1063.

Tárraga, M., Carniel, R., Ortiz, R., García, A., De la Cruz-Reyna, S., 2007. Influence of tectonic events on volcanic activity and implications for pre- eruptive warnings. Geophysical Research Abstracts 9: 02548, European Geosciences Union, General Assembly 2007, Vienna, Austria, 15 - 20 April 2007.

Tárraga, M., Carniel, R., Ortiz, R., García, A., Moreno, H., 2008. A dynamical analysis of the seismic activity of Villarrica volcano (Chile) during September-October 2000. Chaos, Solitons & Fractals 37, 1292—1299.

Tárraga, M., Carniel, R., Ortiz, R., Marrero, J. M., García, A., 2006. On the predictability of volcano-tectonic events by low frequency seismic noise analysis at Teide-Pico Viejo volcanic complex, Canary Islands. Natural Hazards and Earth System Sciences 6, 365—376.

Tjostheim, D., 1981. Identification of seismic sources: Earthquake or underground explosion. in multidimensional discrimination techniques: Theory and application in: Identification of seismic sources — earthquake or underground explosion. In: Proceedings of the NATO Advanced Study Institute held at Voksenasen, Oslo, Norway, September 8-18, 1980. Husebye E. S. Mykkeltveit S. Editors D. Reidel Publishing Company, Boston, Massachusetts.

Trnkoczy, A., 1998. Understanding & setting sta/lta trigger algorithm parameters for the k2. Tech. rep., Kinemetrics, Inc., application Note #41.

Tsuboi, S., Tromp, J., Komatitsch, D., 2004. An XML-SEED Format for the Exchange of Synthetic Seismograms. In: AGU Fall Meeting Abstracts. pp. SF31B–03.

Waible, A., Hanazawa, T., Hinton, G., K.Shikano, Lang, K. J., Mar 1989. Phoneme recognition using time-delay neural networks. IEEE Trans. on Acoustics, Speech, and Signal Processing ASSP-37 (3), 328—339.

Wassermann, J., Beyreuther, M., Ohrnberger, M., Carniel, R., 2007. The usability of hidden markov modelling in seismic detection and automatic warning level estimation of volcanic activity. In: Seismological Research Letters. p. 249.

Weber, B., Becker, J., Hanka, W., Heinloo, A., Hoffmann, M., Kraft, T., Pahlke, D., Reinhardt, J., Thoms, H., 2007. SeisComP3 - automatic and interactive real time data processing. Geophysical Research Abstracts, European Geosciences Union (EGU) General Assembly 9, 09219.

Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., Moore, S., Trujillo, J., 1998. A comparison of selected trigger algorithms for automated global seismic phase and event detection. Bulletin of the Seismological Society of America 88, 95—106.

Young, S., Evermann, G., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P., 2002. The HTK Book. Tech. rep., Cambridge University Engineering Department, HTK Version 3.2.1.

Zen, H., Oura, K., Nose, T., Yamagishi, J., Sako, S., Toda, T., Masuko, T., Black, A. W., Tokuda, K., 2009. Recent development of the hmm-based speech synthesis system (hts). In: Proc. APSIPA. pp. 1–10.

Zen, H., Tokuda, K., Masuko, T., Kobayashi, T., Kitamura, T., 2004. Hidden semi-Markov model based speech synthesis. In: Proc. of ICSLP. Vol. 2. pp. 1397–1400.

A ObsPy: A Python Toolbox for Seismology

by Moritz Beyreuther, Robert Barsch, Lion Krischer, Tobias Megies, Yannik Behr and Joachim Wassermann

Published in Seimological Research Letters, 2010, 81(3), 530-533

A.1 Introduction

The wide variety of computer platforms, file formats and methods to access data in Seismology often requires considerable effort in preprocessing seismic data. Although preprocessing work flows are mostly very similar, hardly any software standards exist to accomplish this task. The objective of ObsPy is to provide a Python toolbox for seismology that simplifies the usage of Python programming for seismologists. It is in that conceptually similar to SEATREE (Milner and Thorsten, 2009) or the exploration seismic software project MADAGASCAR (`http://www.reproducibility.org`).
In ObsPy the following essential seismological processing routines are implemented and ready to use: reading and writing data only SEED/MiniSEED and Dataless SEED `http://www.iris.edu/manuals/SEEDManual_V2.4.pdf`, XML-SEED (Tsuboi et al., 2004), GSE2 `http://www.seismo.ethz.ch/autodrm/downloads/provisional_GSE2.1.pdf` and SAC (`http://www.iris.edu/manuals/sac/manual.html`), as well as filtering, instrument simulation, triggering and plotting. There is also support to retrieve data from ArcLink (a distributed data request protocol for accessing archived waveform data, see Hanka and Kind, 1994) or a SeisHub database (Barsch, 2009). Just recently modules were added to read SEISAN data files (Havskov and Ottemöller, 1999) and to retrieve data with the IRIS/FISSURES Data Handling Interface (DHI) protocol (Malone, 1997).

Python gives the user all the features of a full-fledged programming language including a large collection of scientific open-source modules. ObsPy extends Python by providing direct access to the actual time series, allowing powerful numerical array-programming modules like NumPy (`http://numpy.scipy.org`) or SciPy (`http://scipy.org`) to be used and results to be visualized using modules such as matplotlib (2D) (Hunter, 2007), or MayaVi (3D) (`http://code.enthought.com/projects/mayavi/`). This is an advantage over the most commonly used seismological analysis packages SAC, SEISAN, SeismicHandler (Stammler, 1993) or PITSA

(Scherbaum and Johnson, 1993) which do not provide methods for general numerical array manipulation.
Because Python and its previously mentioned modules are open-source there are no restrictions due to licensing. This is a clear advantage over the proprietary product MATLAB (`http://www.mathworks.com`) in combination with MatSeis (Creager, 1997) or CORAL (Harris and Young, 1997), where the number of concurrent processes is limited by a costly and restricting license policy.
Additionally, Python is known for its intuitive syntax, it is platform independent and has a rapidly growing popularity not only in the seismological community (Olsen and Ely, 2009). Python is used in various fields because of its comprehensive standard library providing tools for all kinds of tasks (e.g. complete web servers can be written in a few lines with standard modules). It has excellent features for wrapping external shared C or FORTRAN libraries, which are used within ObsPy to access libraries for manipulating MiniSEED (libmseed; `http://www.iris.edu/pub/programs`) and GSE2 (gse_util; `http://www.orfeus-eu.org/Software/softwarelib.html#gse`) volumes. Similarly, seismologists may wrap their own C or FORTRAN code and thus are able to quickly develop powerful and efficient software.
In the next section we will give a brief introduction on the capabilities of ObsPy by demonstrating the data conversion of SAC files to MiniSEED volumes, removing the instrument response, applying a low-pass filter and plotting the resulting trace. We then give an overview how to access an external C or FORTRAN library from within Python.

A.2 Reading and Writing

ObsPy provides unified access to read seismograms formated as GSE2, MiniSEED, SAC or SEISAN. For example, entering the following code in a Python shell/interpreter

```
>>> from obspy.core import read
>>> st = read("my_file")
```

automatically detects the file format and loads the data into a stream object which consists of multiple trace objects itself. In MiniSEED as well as GSE2 multiple data records can be stored into one single file. These separate data records are each read into one trace object.
The header attributes of the first trace (`tr = st[0]`) can be addressed by the `tr.stats` object (e.g. `tr.stats.sampling\_rate`). The attribute `tr.data` contains the data as a numpy.ndarray object (array-programming). Thus the data can be further processed by standard Python, NumPy, SciPy, matplotlib or ObsPy routines, e.g. by applying NumPy's fast fourier transform for real valued data:

```
>>> import numpy
>>> print numpy.fft.rfft(tr.data - tr.data.mean())
```

For a conversion from one file format to another the write method of the stream object can be used:

```
>>> from obspy.core import read
>>> st = read("my_file.sac")
>>> st.write("my_file.mseed",format="MSEED")
```

By using the concept of streams and traces we will introduce more functions of ObsPy through the following undergraduate students homework example.

A.3 An Undergraduate-Level Exercise

The task is to extract 30s data via ArcLink from WebDC (`http://webdc.eu`) (Fig. A.1, #1), deconvolve the instrument response and simulate an instrument with 1Hz corner frequency (Fig. A.1, #2). The corrected trace should be low-passed at 10Hz (Fig. A.1, #3) and plotted together with the original seismogram (Fig. A.1, #4 and Fig. A.2).
The example in Fig. A.1 can be extended by fetching data from two different stations, preprocessing them in the same way as in Fig. A.1 and then directly passing the resulting traces to a custom cross-correlation function written in C. The technical details on how to wrap the custom cross-correlation function or any other C or FORTRAN function follows in the next section.

A.4 Extending Python with a Custom Shared Library

In general, interpreters (e.g. Python, Perl, R, MATLAB) are considered slower than compiled source code (e.g. C, FORTRAN). Therefore performance can be optimized by transferring routines with time critical code from Python to compiled shared libraries. Python's foreign function library "ctypes" enabled the ObsPy developers to pass data from Python to functions in shared C or FORTRAN libraries. In doing so the Python memory is accessed directly from the C or FORTRAN function, that is no memory copying is necessary. Further, the complete interface part is written in the interpreter language (no modification of the C code) which makes it easy to reuse existing code. An example of how to access the custom cross correlation C function

```
void X_corr(float *tr1, float *tr2, int param, int ndat1, \
int ndat2, int *shift, double* coe_p)
```

from Python is provided in Fig. A.3 (please notice that a standard cross correlation is also included in SciPy).

In line 1 and 2 of the program (Fig. A.3) the modules and the shared library are loaded (.so stands typically for a shared library on Linux, this may vary for other operating systems). In line 4 and 5 pointers to hold the resulting shift and cross correlation

```
from obspy.core import UTCDateTime
from obspy.arclink import Client
from obspy.signal import cornFreq2Paz, seisSim, lowpass
import numpy as np, matplotlib.pyplot as plt
#1 Retrieve Data via Arclink
client = Client(host="webdc.eu", port=18001)
t = UTCDateTime("2009-08-24 00:20:03")
one_hertz = cornFreq2Paz(1.0) # 1Hz instrument
st = client.getWaveform("BW", "RJOB", "", "EHZ", t, t+30)
paz = client.getPAZ("BW", "RJOB", "", "EHZ", t,
                    t+30).values()[0]
#2 Correct for frequency response of the instrument
res = seisSim(st[0].data.astype("float32"),
st[0].stats.sampling_rate,
paz, inst_sim=one_hertz)
# correct for overall sensitivity, nm/s
res *= 1e9 / paz["sensitivity"]
#3 Apply lowpass at 10Hz
res = lowpass(res, 10, df=st[0].stats.sampling_rate,
corners=4)
#4 Plot the seismograms
sec = np.arange(len(res))/st[0].stats.sampling_rate
plt.subplot(211)
plt.plot(sec,st[0].data, "k")
plt.title("%s %s" % ("RJOB",t))
plt.ylabel("STS-2")
plt.subplot(212)
plt.plot(sec,res, "k")
plt.xlabel("Time [s]")
plt.ylabel("1Hz CornerFrequency")
plt.show()
```

Figure A.1: Advanced Example. Data as well as the instrument response are fetched via ArcLink (please notice the ArcLink server at `http://webdc.eu` is sometimes unreachable). A seismometer with 1Hz corner frequency is simulated and the resulting data are low-passed at 10Hz. An example of the original and the resulting data is plotted and shown in Fig. A.2.

coefficient are allocated and passed to the C function in lines 6, 7, 8 and 9. Note that `data1` and `data2` need to be `numpy.ndarrays` of type "float32" (type checking is included in ctypes but omitted in the example for simplicity). The values of the pointers can now be accessed via the attributes `shift.value` and `coe\_p.value`. At this point we want to emphasize that this interface design differs from MATLAB where the interfaces need to be written in the compiled language (MEX-files).

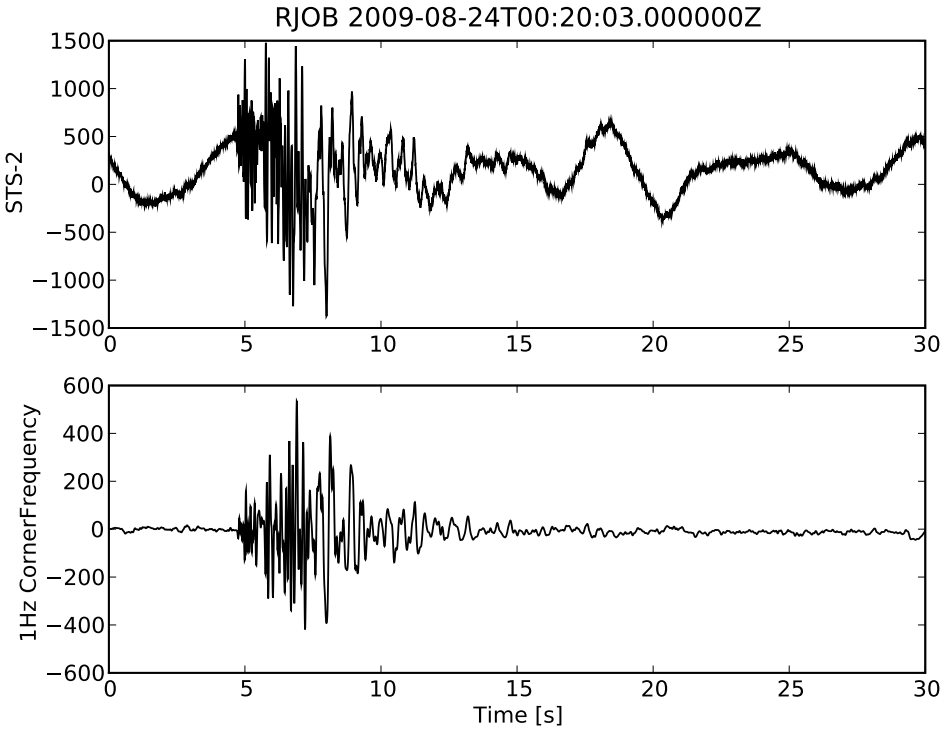

Figure A.2: Local earthquake recorded at station RJOB. STS-2 instrument in the top figure, in the bottom figure the simulated instrument with 1Hz corner frequency low-pass filtered at 10Hz.

```
1 import ctypes as C
2 lib = C.CDLL("pathto"+"xcorr.so")
3 #
4 shift = C.c_int()
5 coe_p = C.c_double()
6 lib.X_corr(data1.ctypes.data_as(C.POINTER(C.c_float)),
7             data2.ctypes.data_as(C.POINTER(C.c_float)),
8             window_len, len(data1), len(data2),
9             C.byref(shift), C.byref(coe_p))
```

Figure A.3: C Interface for accessing the shared library `xcorr.so` from the Python side using ctypes. The resulting shift and correlation coefficient can be accessed via the attributes `shift.value` and `coe_p.value`.

A.5 Discussion and Conclusion

The intention behind ObsPy is not to provide a complete seismological analysis software but to give seismologists access to the basic functionalities they need in order to use Python to easily combine their own programs (Python libraries, modules written

in C/FORTRAN). It could also be seen as a nucleation point for a standard seismology package in Python. Key factors justifying continuation of the ObsPy package are: test-driven development (currently containing 304 tests), modular structure, reliance on well-known third-party tools where possible, open-source code and platform independence (Win, Mac, Linux). The ObsPy package and detailed documentation can be accessed at `http://www.obspy.org`. We encourage any interested user to develop new functions/applications and to make them available via the central repository `https://svn.obspy.org`. We believe that such a free and open-source development environment backed up by Python makes ObsPy a useful toolbox for many seismologists.

A.6 Acknowledgments

We want to thank Heiner Igel and Christian Sippl for their kind support. Further we especially like to thank Chad Trabant, Stefan Stange and Charles J. Ammon whose libraries are the basis for the MiniSEED, GSE2 and SAC support, respectively.
The suggestions of Kim Olsen substantially improved the manuscript.
This work was partially funded by the German Ministry for Education and Research (BMBF), GEOTECHNOLOGIEN grant 03G0646H.

Acknowledgments

An dieser Stelle möchte ich mich bei allen bedanken, die mir bei der Erstellung dieser Doktorarbeit geholfen haben.

In allererster Linie gehört mein Dank Joachim Wassermann für seine uneingeschränkte Unterstützung. Erst durch ihn bin ich mit Hidden Markov Modellen in Berührung gekommen. Mit seinem wertvollen Wissen in der Signalverarbeitung und den daraus resultierenden vielen Diskussionen und Anregungen hat er wesentlich zum Gelingen dieser Arbeit beigetragen.

Auch danke ich meinem Betreuer Heiner Igel. Seine unkomplizierte Art zu Helfen hat viele Probleme schnellstens gelöst. Auch die vielen Diskussionen zur Vorbereitung des seismologischen Signal-Verarbeitungskurses "ROSE" haben mein Wissen in der Signalverarbeitung erweitert.

Bei Matthias Ohrnberger möchte ich mich für seine vielen Tipps und sein bereitwillig geteiltes Experten-Wissen bezüglich Hidden Markov Modellen bedanken. Ohne die Grundlagen seiner Doktorarbeit wäre diese Doktorarbeit nicht möglich gewesen. Auch die Diskussionen mit seinen Wissenschaftlern Conny Hammer und Carsten Riggelsen haben zum Erfolg dieser Doktorarbeit beigetragen.

Robert Barsch, Lion Krischer und Tobias Megies möchte ich für ihre Energie, ihre Motivation und ihren Einsatz für unser "ObsPy: A Python Toolbox for Seismologists" danken. Ohne sie hätte das Projekt bei weitem nicht, wenn überhaupt, den heutigen Stellenwert. Dazu haben auch viele, hier unerwähnte Personen, wertvolle Beiträge geleistet. Bei ihnen allen möchte ich mich für die hervorragende Zusammenarbeit, ihre vielen und wertvollen Beiträge und natürlich die vielen Anregungen bedanken.

Beim gesamten Geophysikalischen Observatorium in Fürstenfeldbruck bedanke ich mich für eine tolle Arbeitsatmosphäre. Besonders die Diskussionen mit Sven Egdorf, Martin Feller und Werner Bauer haben immer wieder für die ebenso nötige Ablenkung gesorgt. Ein ganz besonderer Dank geht dabei an Tobias Megies, der mich insbesondere während des letzten Jahres meiner Doktorarbeit durch sein Engagement und die vielen Diskussionen motiviert hat.

Meiner Tochter Emma danke ich für ihre Art der Unterstützung, diese Doktorarbeit zielgerichtet zu beenden. Mein ganz besonderer Dank gilt meiner lieben Frau Anja, die alle Veröffentlichung gegengelesen hat und mich immer noch erträgt.